



TESIS - TM 142501

**STUDI NUMERIK *SPRING CONSTANT* PADA MATERIAL  
ISOTROPIK MENGGUNAKAN *DISCRETE ELEMENT  
METHOD***

JARIYANTI  
NRP 2114201008

DOSEN PEMBIMBING  
Dr.Eng. Sutikno, ST., MT.

PROGRAM MAGISTER  
BIDANG KEAHLIAN REKAYASA DAN SISTEM MANUFAKTUR  
JURUSAN TEKNIK MESIN  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2017

*Halaman Ini Sengaja Dikosongkan*

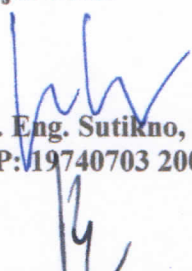
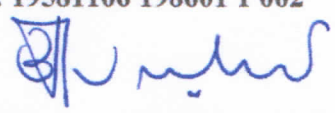
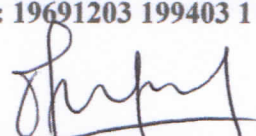
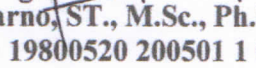
**LEMBAR PENGESAHAN  
TESIS**

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar  
Magister Teknik (M.T)  
di  
Institut Teknologi Sepuluh Nopember

Oleh:  
Jariyanti  
NRP. 2114201008

Tanggal Ujian : Selasa, 17 Januari 2017  
Periode Wisuda : Maret 2017

Disetujui Oleh:

-   
1. Dr. Eng. Sutikno, ST., MT. (Pembimbing)  
NIP: 19740703 200003 1 001
-   
2. Prof. Dr. Ir. I Made Londen Batan., M.Eng. (Penguji)  
NIP: 19581106 198601 1 002
-   
3. Ir. Bambang Pramujati, M.Sc.Eng., Ph.D. (Penguji)  
NIP: 19691203 199403 1 001
-   
4. Suwarno, ST., M.Sc., Ph.D. (Penguji)  
NIP: 19800520 200501 1 003

an, Direktur Program Pascasarjana  
Asisten Direktur



Prof. Dr. Ir. Tri Widjaja, M.Eng.  
NIP: 19611021 198603 1 001

Direktur Program Pascasarjana,

Prof. Ir. Djauhar Manfaat, M.Sc., Ph.D.  
NIP: 196012021987011001

# STUDI NUMERIK *SPRING CONSTANT* PADA MATERIAL ISOTROPIK MENGGUNAKAN *DISCRETE ELEMENT METHOD*

Mahasiswa : Jariyanti  
NRP : 2114201008  
Pembimbing : Dr. Eng. Sutikno, ST., MT

## ABSTRAK

*Discrete Element Method* (DEM) merupakan salah satu metode numerik yang digunakan untuk pemodelan dan analisis. Dalam analisis DEM, obyek diasumsikan sebagai kumpulan elemen diskrit dan interaksi diantara elemen ditentukan oleh konstanta pegas, kelembaman dan koefisien gesek. Pemodelan konstanta pegas pada arah normal ( $k_n$ ) dan tangensial ( $k_s$ ) telah banyak diteliti dengan pendekatan jumlah elemen dalam *cluster* dan ukuran elemen yang *uniform*. Pendekatan ukuran elemen yang *uniform* sering kali tidak dapat digunakan untuk bentuk geometri yang kompleks, oleh karena itu, diperlukan pendekatan yang lain. Pada penelitian ini, akan dianalisa secara numerik fungsi konstanta pegas pada arah normal ( $k_n$ ) dan tangensial ( $k_s$ ) dengan pendekatan dua elemen yang memiliki ukuran (radius) yang berbeda.

Pada penelitian ini, material yang dimodelkan adalah material isotropik. Bentuk geometri yang dimodelkan adalah heksagonal dengan asumsi didalamnya terdapat lingkaran yang menyinggung tiap sisi, dengan ukuran radius ( $r_i$ ) pada elemen  $i$  dan radius ( $r_j$ ) pada elemen  $j$ . Selanjutnya, menentukan interaksi gaya antar elemen, kemudian menentukan komponen tegangan. Komponen tegangan yang diperoleh dari pemodelan, dibandingkan dengan matriks konstitutif material isotropik untuk menentukan fungsi  $k_n$  dan  $k_s$ .

Hasil penelitian menunjukkan bahwa nilai *displacement* dari simulasi fungsi  $k_n$  dan  $k_s$ , serta analitik masing-masing sebesar  $-0.000556$  mm dan  $-0.000585$  mm. Nilai *error* perbandingan hasil simulasi dan analitik sebesar 4.89486%.

**Kata kunci:** *Discrete element method* (DEM), matriks konstitutif, *isotropic*, konstanta pegas.

*(halaman ini sengaja dikosongkan)*

# NUMERICAL STUDY OF SPRING CONSTANT ON ISOTROPIC MATERIAL USING DISCRETE ELEMENT METHOD

Name of Student : Jariyanti  
Student ID : 2114201008  
Supervisor Lecture : Dr. Eng. Sutikno, ST., MT

## ABSTRACT

Discrete Element Method (DEM) is one of the numeric method used to modeling and analysis. The object of DEM analysis is assumed as a collection of discrete element and the interaction among the elements is determined by the spring constant, inertia and friction coefficient. Modeling Spring constant in normal ( $k_n$ ) and tangential direction ( $k_s$ ) has been studied using the number of elements in cluster approach and the size of the uniform elements. The approach of uniform element size oftentimes cannot be used to complex geometric forms. Therefore, another approach is needed.

This study aims to analyze spring constant formula numerically in normal ( $k_n$ ) and tangential ( $k_s$ ) direction with the approach of two elements that have different size (radius). Material modeling is an isotropic material. Geometric modeling is a hexagon. Assume the circle inscribed into a hexagon with  $r_i$  is the size of the radius on element  $i$  and  $r_j$  is size of the radius on element  $j$ . Furthermore, determine the interaction of force between elements, then determine the stress components. Compared Stress components from modeling with the constitutive matrix of isotropic material to determine the function  $k_n$  and  $k_s$ .

The results shown that the displacement values of  $k_n$  and  $k_s$  simulation and analytical are -0.000556 mm and -0.000585 mm with a ratio value of the error function of  $k_n$  and  $k_s$  is 4.89486%.

**Key word:** Discrete element method (DEM), the constitutive matrix, isotropic, spring constant.

*(halaman ini sengaja dikosongkan)*

## KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah memberi rahmat, hidayah serta anugrah-Nya sehingga penulis dapat menyelesaikan tesis ini dengan lancar. Salawat dan salam semoga tetap tercurahkan kepada Nabi Muhammad SAW. Penulisan tesis dengan judul **“Studi Numerik *Spring Constant* pada Material Isotropik Menggunakan *Discrete Element Method*”** telah selesai dilaksanakan. Tesis ini merupakan salah satu syarat yang harus ditempuh untuk menyelesaikan pendidikan guna memperoleh gelar Magister Teknik (M.T) di Jurusan Teknik Mesin, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya tahun 2017.

Penulis menyadari selama penyusunan tesis ini telah melibatkan banyak pihak yang sangat membantu. Oleh karena itu, penulis menyampaikan penghargaan dan ucapan terima kasih yang sebesar-besarnya kepada:

1. Bapak Ir. Bambang Pramujati, M.Sc.Eng., Ph.D, selaku Ketua Jurusan Teknik Mesin, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember dan jajarannya.
2. Bapak Dr. Eng. Sutikno, ST., MT, selaku dosen pembimbing yang selalu memberi nasehat, bimbingan dan motivasi.
3. Bapak Prof. Dr. Ing. Ir. I Made Londen Batan, M.Eng, Bapak Ir. Bambang Pramujati, M.Sc.Eng., Ph.D dan Bapak Suwarno, ST., M.Sc., Ph.D. selaku dosen penguji yang telah memberi saran yang bermanfaat kepada penulis.
4. Bapak Prof. Dr. Eng. Prabowo, M.Eng. selaku Kaprodi Pascasarjana Jurusan Teknik Mesin, FTT-ITS, yang telah memberi arahan dan kemudahan kepada penulis dalam penyelesaian tesis.
5. Pihak Pascasarjana Institut Teknologi Sepuluh Nopember Surabaya yang telah memberikan kesempatan penulis dalam mengenyam pendidikan Magister di Teknik Mesin lewat pemberian program beasiswa *Pra S2-S2 Saintek 3T* di Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2013.
6. Orang tua penulis, Ayahanda La Toni dan Ibunda Jarniah yang telah memberikan curahan kasih sayang, dukungan, semangat, motivasi dan do'a kepada penulis.



7. Teman-teman Teknik Mesin, khususnya Bidang Rekayasa dan Sistem Manufaktur, Moh. Muzaki, Sufiyanto, Ali Sai'in, Thenny, Benedictus, Firman, Balkhaya, Husnul Abid, Muhamad Anhar Pulungan, Hiding dan Faisal Manta yang telah sama-sama berjuang dalam menuntut ilmu.
8. Teman-teman Teknik Mesin Bidang Rekayasa Konversi Energi, Alfi Tranggono, Agus Choirul Arifin, Indarto, Luthfi, Izhari, Romy Djafar, Sulaiman Ali dan Masrur atas kebersamaannya.
9. Teman-teman Teknik Material dan Metalurgi, Fahriadi Pakaya, Saddam Husen, Mustofa, Yulianti Malik dan Nia Sasria yang selalu memberikan semangat dan dukungan.
10. Seluruh karyawan Jurusan Teknik Mesin yang banyak membantu dalam penyelesaian pengerjaan tesis ini.
11. Semua pihak yang tidak dapat penulis sebutkan satu-persatu yang telah banyak memberi dorongan dan bantuan dalam menyelesaikan tesis ini baik secara langsung maupun tidak langsung.

Penulis menyadari masih banyak kekurangan dan keterbatasan pengetahuan dalam penelitian sehingga dimungkinkan ada kekeliruan dan kesalahan yang tidak sengaja. Oleh karena itu kritik dan saran yang membangun sangat dibutuhkan guna perbaikan dan pengembangan lebih lanjut. Semoga tesis dapat bermanfaat dan memenuhi apa yang diharapkan.

Surabaya, Januari 2017

Penulis

## DAFTAR ISI

	Hal
<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>LEMBAR PENGESAHAN .....</b>	<b>iii</b>
<b>ABSTRAK .....</b>	<b>v</b>
<b>ABSTRACT .....</b>	<b>vii</b>
<b>KATA PENGANTAR .....</b>	<b>ix</b>
<b>DAFTAR ISI .....</b>	<b>xi</b>
<b>DAFTAR GAMBAR .....</b>	<b>xiii</b>
<b>DAFTAR TABEL .....</b>	<b>xv</b>
<b>BAB 1. PENDAHULUAN</b>	
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Penelitian .....	3
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	3
<b>BAB 2. TINJAUAN PUSTAKA</b>	
2.1 <i>Discrete Element Method</i> (DEM) .....	5
2.2 Tegangan .....	7
2.3 Regangan.....	8
2.4 Hukum Hooke .....	9
2.5 Penelitian Terdahulu .....	10
<b>BAB 3. METODOLOGI PENELITIAN</b>	
3.1 Diagram Alir Penelitian .....	15
3.2 Tahapan Penelitian .....	17
3.3 Data .....	17
3.4 Formulasi Numerik .....	17
3.5 Simulasi DEM.....	19
3.6 Analisa Hasil .....	21
<b>BAB 4. HASIL DAN PEMBAHASAN</b>	
4.1 Formulasi <i>spring constant</i> arah normal dan tangensial .....	23
4.2 Hasil Simulasi dengan Menggunakan <i>Spring Constant</i> yang Diperoleh Gaeni .....	28
4.3 Hasil Simulasi dengan Menggunakan <i>Spring Constant</i> yang Diperoleh dari Pemodelan .....	29

4.4 Hasil Simulasi dengan Menggunakan <i>Spring Constant</i> yang Diperoleh dari Pemodelan dengan Jumlah 177 Elemen .....	30
4.5 Hasil Perbandingan <i>Displacement</i> Antara Simulasi dengan <i>Spring Constant</i> Gaeni dan <i>Spring Constant</i> dari Formulasi .....	31
4.6 Hasil Analisis <i>Displacement</i> pada Simulasi dengan Jumlah Elemen yang Berbeda.....	33
4.7 Perbandingan Hasil Analisis Defleksi Spesimen.....	34

## **BAB 5. KESIMPULAN DAN SARAN**

5.1 Kesimpulan .....	37
5.2 Saran .....	37

## **DAFTAR PUSTAKA .....39**

## **LAMPIRAN**

Lampiran 1. Coding Simulasi DEM .....	41
---------------------------------------	----

## DAFTAR GAMBAR

	Hal
Gambar 1.1 Bentuk dan Susunan Elemen .....	2
Gambar 2.1 Diagram Kontak Antara Elemen $i$ dan $j$ .....	5
Gambar 2.2 (a) Gaya Lurus Bidang (b) Gaya Sejajar Bidang .....	7
Gambar 2.3 (a) Tiga Dimensi (b) Dua Dimensi .....	8
Gambar 2.4 Susunan Tujuh Elemen .....	10
Gambar 2.5 Susunan Tujuh Elemen .....	10
Gambar 2.6 Perbandingan <i>Displacement</i> DEM dan FEA.....	11
Gambar 2.7 Susunan Tujuh Elemen .....	11
Gambar 2.8 Perbandingan Tegangan DEM dan LS-DYNA .....	12
Gambar 2.9 Susunan Dua Elemen .....	12
Gambar 2.10 Perbandingan Tegangan dengan Tujuh Elemen dan Dua Elemen	13
Gambar 3.1 Diagram Alir Metodologi Penelitian .....	15
Gambar 3.2 Diagram Alir Simulasi DEM.....	16
Gambar 3.3 DEM <i>Calculation Cycle</i> .....	16
Gambar 3.4 Dimensi Material .....	20
Gambar 3.5 Susunan Elemen pada Simulasi .....	20
Gambar 4.1 Geometri dan Ukuran pada Elemen $i$ dan $j$ .....	23
Gambar 4.2 Gaya pada Elemen $i$ dan $j$ .....	23
Gambar 4.3 Simulasi dengan <i>Spring Constant</i> yang Diperoleh Gaeni dengan Jumlah Elemen 173 .....	28
Gambar 4.4 Simulasi dengan <i>Spring Constant</i> yang diperoleh pada Penelitian ini dengan Jumlah elemen 173 .....	29
Gambar 4.5 Simulasi dengan <i>spring constant</i> yang diperoleh pada penelitian ini dengan Jumlah Elemen 177 .....	30
Gambar 4.6 Grafik Perbandingan Simulasi Menggunakan <i>Spring Constant</i> yang Diperoleh dari formulasi dan <i>Spring Constant</i> dari Formulasi Gaeni .....	32
Gambar 4.7 Grafik Perbandingan <i>Displacement</i> Kearah $y$ dengan Jumlah Elemen 173 dan 177 .....	33
Gambar 4.8 Dimensi dan Pemberian Gaya .....	35

*Halaman ini sengaja dikosongkan*

## DAFTAR TABEL

	Hal
Tabel 3.1 Properti material dan parameter DEM .....	17
Tabel 4.1 Perhitungan <i>error</i> dari perbandingan perhitungan analitik dan simulasi .....	35

*Halaman ini sengaja dikosongkan*

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Pemodelan merupakan salah satu cara pendekatan yang sering digunakan dalam bidang rekayasa dan teknik. Pada prinsipnya, pemodelan menyelesaikan masalah fisik dengan cara penyederhanaan suatu sistem. Secara umum, pemodelan dibagi menjadi dua yaitu metode analitik dan metode numerik. Untuk suatu sistem yang kompleks, metode analitik sulit untuk digunakan, sehingga perlu menggunakan metode numerik.

Metode numerik adalah suatu metode yang digunakan untuk menyelesaikan berbagai permasalahan yang dapat diformulasikan secara matematis dengan menggunakan cara operasi hitungan (*arithmetic*). Metode ini mampu menyelesaikan sistem persamaan yang besar, tidak linier dan geometri rumit yang jarang terjadi dalam praktik keteknikan serta seringkali sulit diselesaikan dengan cara analitis. Adapun metode numerik yang sering digunakan dalam rekayasa dan teknik adalah *finite difference method* (FDM), *boundary element method* (BEM), *finite element method* (FEM) dan *discrete element method* (DEM).

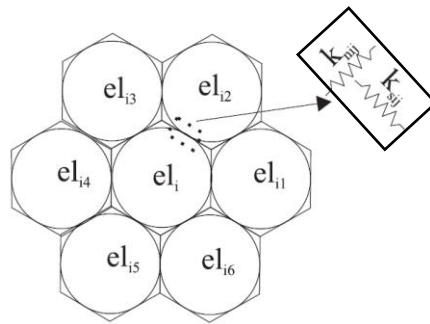
*Discrete Element Method* (DEM) merupakan salah satu metode numerik yang diperkenalkan pertama kali oleh Cundall dan Strack (1979) untuk analisa granular. Analisis objek dalam DEM diasumsikan sebagai kumpulan partikel diskrit dan interaksi mekanis diantara partikel ditentukan oleh konstanta pegas (konstanta pegas normal ( $k_n$ ) dan konstanta pegas geser ( $k_s$ )), konstanta daspot atau kelembaman dan konstanta gesek, Purwantana (2004). Konstanta pegas menentukan *displacement* yang terjadi pada partikel yang diberi beban atau gaya. Ketika gaya yang diberikan melebihi batas elastisitasnya, maka partikel tersebut akan terpisah dari kumpulan partikel-partikel lainnya.

Pemodelan fungsi  $k_n$  dan  $k_s$  menggunakan DEM dalam struktur beton diteliti oleh Sawamoto dkk (1998). Pendekatan yang digunakan yaitu tujuh elemen dengan susunan satu elemen dikelilingi oleh enam elemen lainnya. Hasil penelitian menunjukkan bahwa model DEM sangat akurat untuk menilai kerusakan lokal pada



struktur beton yang diberi beban tumbukan. Pendekatan sejenis diteliti oleh Tavaréz dkk (2007), yang memodelkan fungsi  $k_n$  dan  $k_s$  pada *solid* dan *particulate material*. Penelitian menunjukkan bahwa hasil konvergen diperoleh dari DEM.

Model fungsi  $k_n$  dan  $k_s$  dengan DEM juga diteliti oleh Liu dkk (2004) pada material *orthotropic* untuk masalah *impact*. Bentuk elemen yang dimodelkan adalah heksagonal dengan pendekatan tujuh elemen. Bentuk dan susunan elemen dapat ditunjukkan pada Gambar 1.1. Hasil penelitian menunjukkan bahwa model yang diusulkan dapat digunakan untuk analisis numerik pada masalah *impact*.



Gambar 1.1 Bentuk dan Susunan Elemen (Gaeni, 2008)

Pemodelan pada material isotropik didasarkan dari konsep *discrete element*, diteliti oleh Griffiths dkk (2001). Pendekatan yang digunakan yaitu tiga elemen dengan bentuk dan ukuran yang sama. Hasil penelitian menunjukkan bahwa *displacement* yang diperoleh dari formulasi *discrete element* dan *solid finite element* memiliki nilai yang sama.

Pemodelan pada material isotropik diteliti oleh Gaeni dkk (2008). Pendekatan yang digunakan yaitu dua elemen dengan bentuk heksagonal dan ukuran yang sama. Hasil penelitian menunjukkan bahwa hasil analisis numerik yang diperoleh dari model yang menggunakan dengan model *tujuh-disk*, memiliki hasil yang hampir sama.

Fungsi *spring constant* yang diperoleh oleh Gaeni diimplementasikan dalam masalah *impact*. Hasil penelitian menunjukkan analisis yang diperoleh sama dengan analisis yang diperoleh dengan metode LS-DYNA, Sutikno dkk (2009). Implementasi *spring constant* pada masalah tumbukan partikel pada balok keramik

diteliti Sutikno dkk (2010). Hasil penelitian menunjukkan metode tersebut sangat akurat untuk analisis masalah tumbukan partikel pada balok keramik.

Pada penelitian ini akan dimodelkan  $k_n$  dan  $k_s$  dengan DEM pada material isotropik menggunakan pendekatan dua elemen dengan ukuran yang berbeda.

## **1.2 Perumusan Masalah**

Pemodelan fungsi  $k_n$  dan  $k_s$  sudah banyak diteliti seperti yang telah dipaparkan pada latar belakang. Model  $k_n$  dan  $k_s$  yang diteliti masih berdasarkan jumlah, bentuk dan ukuran elemen yang uniform. Ukuran elemen yang sama dengan susunan 7 elemen kurang akurat jika digunakan sebagai pendekatan pada geometri tertentu khususnya pada bagian tepi material. Oleh karena itu akan diteliti bagaimana formulasi fungsi  $k_n$  dan  $k_s$  menggunakan DEM dengan pendekatan dua elemen heksagonal dengan ukuran yang berbeda pada material isotropik.

## **1.3 Batasan Penelitian**

Batasan penelitian yang digunakan agar penelitian ini terarah dan fokus untuk mencapai tujuan yang diharapkan. Adapun batasan pada penelitian ini adalah damping dan pengaruh gaya gesek antar permukaan elemen diasumsikan tidak ada.

## **1.4 Tujuan Penelitian**

Tujuan penelitian ini yaitu formulasi fungsi *spring constant* pada material isotropik dengan pendekatan dua elemen heksagonal yang memiliki ukuran (radius) yang berbeda.

## **1.5 Manfaat Penelitian**

Manfaat yang diperoleh dari penelitian ini yaitu dapat menambah referensi tentang formulasi *spring constant* dengan DEM.

*(halaman ini sengaja dikosongkan)*

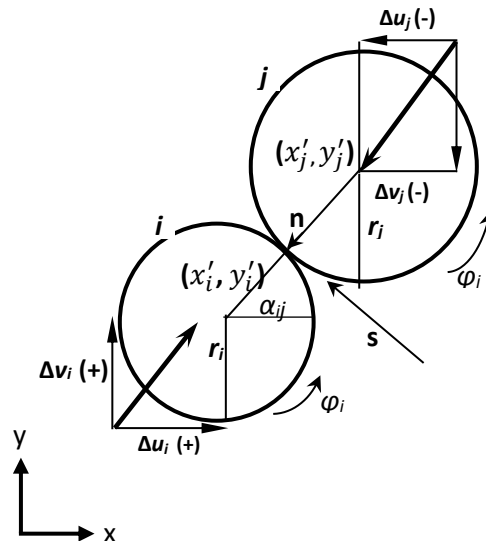
## BAB 2

### KAJIAN PUSTAKA

#### 2.1 *Discrete Element Method (DEM)*

Pemodelan menggunakan DEM, suatu persamaan diberikan pada tiap elemen untuk menganalisis *displacement* yang terjadi. Metode menjabarkan persamaan dalam DEM, gaya-gaya yang bekerja pada masing-masing elemen harus diketahui. Konsep utama DEM yaitu mengasumsikan bahwa obyek yang akan dianalisa merupakan sekumpulan elemen kecil. Setiap elemen menerima gaya kontak dari elemen-elemen disekelilingnya. Besarnya gaya kontak ditentukan berdasarkan *displacement*, dan kecepatan relatif setiap elemen.

Gaya yang bekerja pada setiap elemen sebanding dengan jarak perpindahan dan kecepatan relatif elemen terhadap elemen disekelilingnya yang terjadi kontak dengan elemen tersebut. Ilustrasi ditunjukkan pada Gambar 2.1, yang merupakan dua elemen  $i$  dan  $j$  saling kontak pada waktu  $t$ . Besar dan arah pergerakan setiap elemen selama interval waktu ( $\Delta t$ ) ditunjukkan dengan tanda panah. Perpindahan ke arah horisontal (arah  $x$ ) dan vertikal (arah  $y$ ) elemen  $i$  dan  $j$  selama  $\Delta t$  masing-masing dinyatakan dengan notasi  $\Delta u_i$ ,  $\Delta v_i$ ,  $\Delta u_j$  and  $\Delta v_j$ .



Gambar 2.1 Diagram kontak antara elemen  $i$  dan  $j$

Gaya pada arah normal dan tangensial dapat dinyatakan dengan persamaan-persamaan:

$$[f_n]_t = [f_n]_{t-\Delta t} + k_n \cdot \Delta u_n \quad (2.1)$$

$$[f_s]_t = [f_s]_{t-\Delta t} + k_s \cdot \Delta u_s \quad (2.2)$$

dengan  $f_n$  adalah gaya arah normal,  $f_s$  adalah gaya arah tangensial,  $k_n$  dan  $k_s$  adalah konstanta pegas antara elemen  $i$  dan  $j$  pada arah normal dan tangensial,  $\Delta u_n$  dan  $\Delta u_s$  adalah *displacement* arah normal dan tangensial.

Transformasi rotasi, gaya pada arah normal, dan tangensial dari sistem koordinat lokal menjadi sistem koordinat global, serta menjumlahkan gaya-gaya kontak yang disebabkan oleh semua elemen yang kontak dengan elemen  $i$ , maka diperoleh resultan gaya dan momen sebagai berikut:

$$[Fx_i]_t = \sum_j (-[f_n]_t \cos \alpha_{ij} + [f_s]_t \sin \alpha_{ij}) \quad (2.3)$$

$$[Fy_i]_t = \sum_j (-[f_n]_t \sin \alpha_{ij} - [f_s]_t \cos \alpha_{ij}) \quad (2.4)$$

$$[M_i]_t = -r_i \sum_j ([f_s]_t) \quad (2.5)$$

dengan  $Fx_i$  adalah resultan gaya elemen  $i$  pada arah x,  $Fy_i$  adalah resultan gaya elemen  $i$  pada arah y,  $M_i$  adalah momen pada elemen  $i$ ,  $r_i$  adalah radius pada elemen  $i$ .

Besarnya percepatan pada elemen  $i$ , diperoleh dengan membagi gaya kontak untuk setiap arah dengan masa atau momen inersia dari elemen  $i$  sebagai berikut:

$$[\ddot{u}_i]_t = \frac{[Fx_i]_t}{m_i} \quad (2.6)$$

$$[\ddot{v}_i]_t = \frac{[Fy_i]_t}{m_i} \quad (2.7)$$

$$[\ddot{\phi}_i]_t = \frac{[M_i]_t}{I_i} \quad (2.8)$$

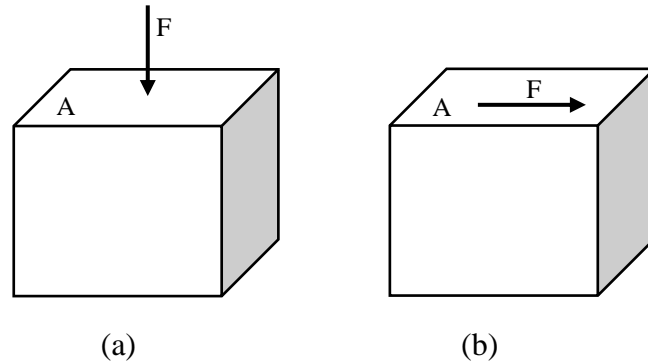
dengan  $\ddot{u}_i$  adalah percepatan elemen  $i$  kearah u,  $\ddot{v}_i$  adalah percepatan elemen  $i$  kearah v, dan  $\ddot{\phi}_i$  adalah percepatan tangensial.

Untuk menentukan kecepatan pada elemen  $i$  diwaktu  $t$ , diperoleh dengan mengintegalkan percepatan untuk setiap selang waktu ( $\Delta t$ ) dan menjumlahkan

nilainya dengan nilai kecepatan sebelumnya. Selanjutnya, dengan cara yang sama yaitu mengintegrasikan kecepatan untuk setiap selang waktu ( $\Delta t$ ) dan menjumlahkan nilainya dengan nilai yang sebelumnya, maka diperoleh *displacement* pada elemen  $i$ .

## 2.2 Tegangan

Tegangan didefinisikan sebagai gaya yang diberikan pada suatu benda dengan luas bidang tertentu. Tegangan dibagi menjadi dua yaitu, tegangan normal ( $\sigma$ ) dan tegangan geser ( $\tau$ ).



Gambar 2.2: (a) Gaya tegak lurus bidang; (b) Gaya sejajar bidang

Tegangan normal adalah gaya yang diberikan pada suatu bidang yang arahnya tegak lurus bidang tersebut. Sedangkan tegangan geser adalah gaya yang diberikan pada suatu bidang yang arahnya sejajar bidang tersebut. Persamaan tegangan normal dan tegangan geser sebagai berikut:

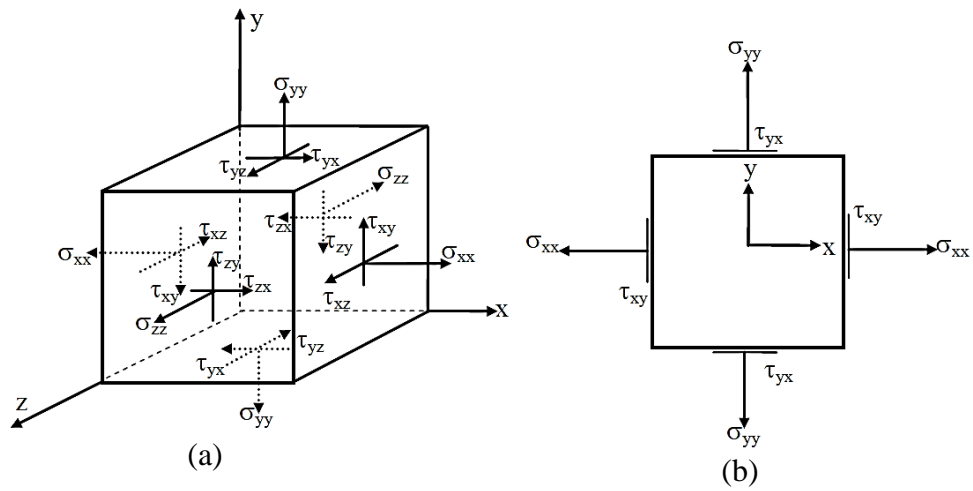
$$\sigma = \frac{F}{A} \quad (2.9)$$

$$\tau = \frac{F}{A} \quad (2.10)$$

dengan  $\sigma$  adalah tegangan normal,  $\tau$  adalah tegangan geser,  $F$  adalah gaya, dan  $A$  adalah luas bidang.

Tensor ialah besaran fisik yang keadaannya pada suatu titik dalam ruang, tiga dimensi, dapat dideskripsikan dengan  $3^n$  komponennya, dengan  $n$  ialah derajat tensor tersebut. Dengan demikian, persoalan tegangan tiga dimensi pada suatu titik dalam ruang dapat dideskripsikan dengan 32 komponennya. Pada sistem koordinat

sumbu silang, tegangan tersebut adalah  $\sigma_{xx}$ ,  $\sigma_{yy}$ ,  $\sigma_{zz}$ ,  $\tau_{xy}$ ,  $\tau_{yx}$ ,  $\tau_{xz}$ ,  $\tau_{zx}$ ,  $\tau_{yz}$ , dan  $\tau_{zy}$  seperti ditunjukkan pada Gambar 2.3 (a). Namun, karena  $\tau_{xy} = \tau_{yx}$ ,  $\tau_{xz} = \tau_{zx}$  dan  $\tau_{yz} = \tau_{zy}$ , maka keadaan tegangan dapat dinyatakan dengan enam komponennya,  $\sigma_{xx}$ ,  $\sigma_{yy}$ ,  $\sigma_{zz}$ ,  $\tau_{xy}$ ,  $\tau_{xz}$ ,  $\tau_{yz}$ , sedangkan tegangan bidang, dua dimensi, pada suatu titik dapat dideskripsikan dengan  $2^2$  komponennya, Gambar 2.3 (b), dan karena  $\tau_{ij} = \tau_{ji}$  untuk  $i \neq j$  maka tiga komponen telah mendeskripsikan tegangan bidang pada titik itu.



Gambar 2.3: (a) Tiga dimensi; (b) Dua dimensi

Jika persamaan (2.9) dituliskan sebagai tensor tegangan, maka persamaan (2.9) dapat ditulis sebagai berikut:

$$\sigma_{ij} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix} \quad (2.11)$$

## 2.3 Regangan

Regangan didefinisikan sebagai nilai perbandingan antara perubahan panjang terhadap panjang awal dengan dimensi yang sama (Dieter, 1988). Secara matematik persamaan regangan adalah:

$$\varepsilon = \frac{\Delta l}{L_0} = \frac{l_t - l_0}{l_0} \quad (2.12)$$

dengan  $\varepsilon$  adalah regangan,  $l_0$  adalah panjang mula-mula, dan  $l_t$  adalah panjang setelah diberi gaya.

Selain tegangan, regangan juga merupakan tensor derajat dua. Dengan demikian, persamaan (2.12) dapat ditulis sebagai berikut:

$$\varepsilon_{ij} = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & \varepsilon_{33} \end{bmatrix} \quad (2.13)$$

## 2.4 Hukum Hooke

Semua benda padat jika diberikan gaya akan mengalami deformasi. Selanjutnya, jika gaya dihilangkan maka akan kembali ke ukuran awal. Perolehan kembali ukuran awal setelah gaya dihilangkan disebut perilaku elastik (*elastic behavior*). Apabila gaya yang diberikan melebihi batas elastik, maka benda akan mengalami deformasi permanen. Benda yang mengalami deformasi permanen (regangan permanen) disebut deformasi plastik.

Deformasi yang terjadi sebanding dengan gaya yang diberikan, hal ini disebut sebagai hukum Hooke. Hukum Hooke lebih sering dinyatakan dalam bentuk hubungan tegangan-regangan. Secara matematik dapat dituliskan menjadi persamaan (2.14)

$$\sigma = E \varepsilon \quad (2.14)$$

dengan  $\sigma$  adalah tegangan,  $E$  adalah modulus Young, dan  $\varepsilon$  adalah regangan.

Hubungan tegangan, regangan, modulus Young dan *poisson ratio* pada material isotropik dapat dituliskan dalam bentuk persamaan berikut:

$$\sigma_{11} = \frac{E}{(1 - \nu^2)} \varepsilon_{11} + \frac{E\nu}{(1 - \nu^2)} \varepsilon_{22} \quad (2.15)$$

$$\sigma_{11} = \frac{E\nu}{(1 - \nu^2)} \varepsilon_{11} + \frac{E}{(1 - \nu^2)} \varepsilon_{22} \quad (2.16)$$

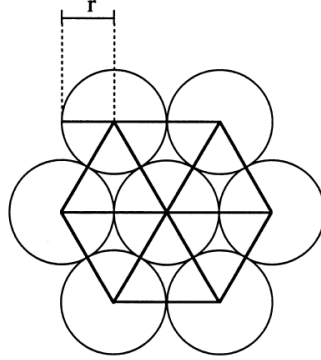
$$\sigma_{12} = \frac{E}{2(1 + \nu)} \varepsilon_{12} \quad (2.17)$$

dengan  $\nu$  adalah *Poisson ratio*. *Poisson ratio* adalah nilai perbandingan regangan lateral dan regangan aksial.



## 2.5 Penelitian Terdahulu

Penelitian tentang formulasi  $k_n$  dan  $k_s$  telah banyak dilakukan. Sawamoto (1998) memformulasikan  $k_n$  dan  $k_s$  dengan pendekatan tujuh elemen, dimana satu elemen dikelilingi enam elemen lainnya seperti terlihat pada Gambar 2.4.



Gambar 2.4 Susunan tujuh elemen (Sawamoto, 1998)

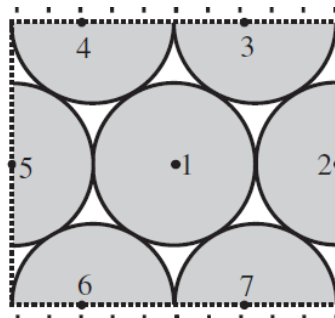
Formulasi  $k_n$  dan  $k_s$  yang diperoleh pada penelitian tersebut adalah:

$$k_n = \frac{\sqrt{3}E}{3(1-\nu)} \quad (2.18)$$

$$k_s = \frac{\sqrt{3}(1-3\nu)E}{3(1-\nu)^2} \quad (2.19)$$

Hasil penelitian menunjukkan bahwa pemodelan dengan DEM sangat akurat untuk menilai kerusakan lokal pada struktur beton yang diberi beban tumbukan.

Pendekatan yang sama juga diteliti oleh Tavaréz (2007), dengan jumlah elemen tujuh elemen, satu elemen dikelilingi setengah elemen. Gambar 2.5 menunjukkan susunan elemen yang digunakan untuk memformulasikan  $k_n$  dan  $k_s$ .



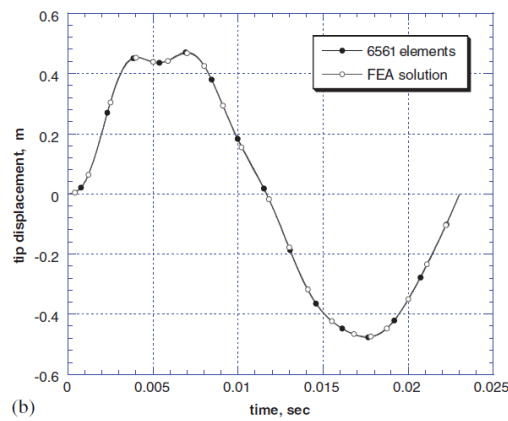
Gambar 2.5 Susunan tujuh elemen (Tavaréz, 2007)

Formulasi  $k_n$  dan  $k_s$  yang diperoleh dengan pendekatan tersebut adalah:

$$k_n = \frac{1}{\sqrt{3}(1-v)} E\delta \quad (2.20)$$

$$k_s = \frac{(1-3v)}{\sqrt{3}(1-v^2)} E\delta \quad (2.21)$$

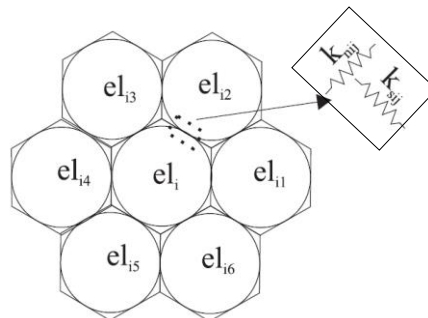
Formulasi yang telah diperoleh kemudian divalidasi untuk mengetahui keakuratan formulasi tersebut. Validasi dilakukan dengan cara simulasi. Hasil yang diperoleh dari simulasi dengan DEM, kemudian dibandingkan dengan hasil dari simulasi dengan FEA. Perbandingan hasil simulasi DEM dan FEA ditunjukkan pada Gambar 2.6.



Gambar 2.6 Perbandingan *displacement* DEM dan FEA (Tavarez, 2007)

Hasil penelitian tersebut menunjukkan bahwa *displacement* yang diperoleh dengan DEM sama dengan *displacement* dengan FEA.

Liu (2004), memformulasikan  $k_n$  dan  $k_s$  dengan pendekatan tujuh elemen. Perbedaan dengan penelitian sebelumnya adalah bentuk elemen yang digunakan heksagonal. Susunan elemen pada penelitian tersebut seperti ditunjukkan pada Gambar 2.7.



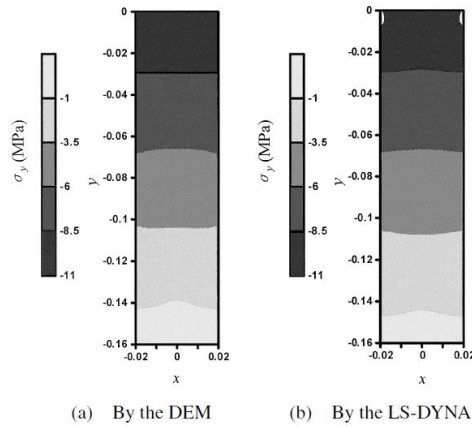
Gambar 2.7 Susunan tujuh elemen (Gaeni, 2008)

Pada penelitian tersebut diperoleh formulasi  $k_n$  dan  $k_s$  sebagai berikut:

$$k_n = \frac{\sqrt{3}}{3} \cdot \frac{E\delta}{1-\nu} \quad (2.22)$$

$$k_s = \frac{\sqrt{3}}{3} \cdot \frac{(1-3\nu)E\delta}{1-\nu^2} \quad (2.23)$$

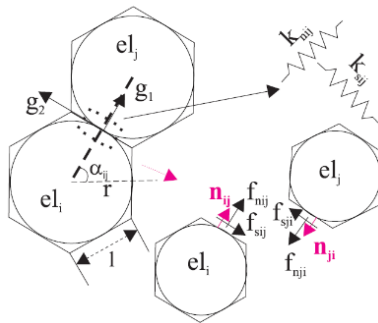
Formulasi yang diperoleh kemudian divalidasi dengan cara simulasi. Hasil simulasi yang diperoleh, dibandingkan dengan hasil LS-DYNA. Perbandingan hasil simulasi DEM dan LS-DYNA ditunjukkan pada Gambar 2.8.



Gambar 2.8 Perbandingan tegangan DEM dan LS-DYNA (Liu, 2004)

Hasil penelitian tersebut menunjukkan nilai tegangan yang diperoleh dengan DEM sama dengan nilai tegangan yang diperoleh dengan LS-DYNA.

Gaeni (2008), memformulasikan  $k_n$  dan  $k_s$  dengan pendekatan dua elemen heksagonal. Susunan elemennya seperti terlihat pada Gambar 2.9.



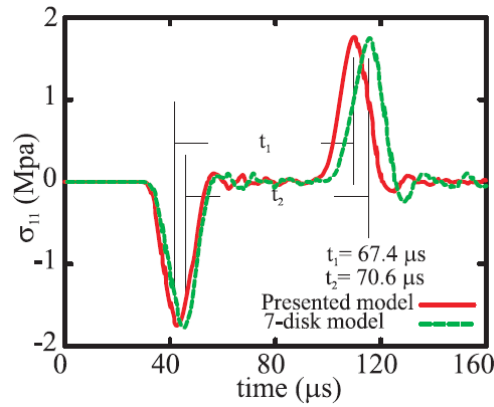
Gambar 2.9 Susunan dua elemen (Gaeni, 2008)

Fungsi  $k_n$  dan  $k_s$  yang diperoleh pada penelitian tersebut adalah:

$$k_n = \frac{\sqrt{3}}{3} \cdot \frac{E\delta}{1-\nu} \quad (2.24)$$

$$k_s = \frac{E\delta}{4(1+\nu)} \quad (2.25)$$

Formulasi yang diperoleh kemudian divalidasi dengan cara simulasi. Hasil simulasi yang diperoleh, dibandingkan dengan hasil simulasi menggunakan formulasi  $k_n$  dan  $k_s$  liu. Perbandingan hasil simulasi ditunjukkan pada Gambar 2.10



Gambar 2.10 Perbandingan tegangan dengan 7 elemen dan dua elemen (Gaeni, 2008)

Hasil penelitian menunjukkan bahwa nilai tegangan yang diperoleh dengan model tujuh elemen, memiliki hasil yang hampir sama dengan model dua elemen.

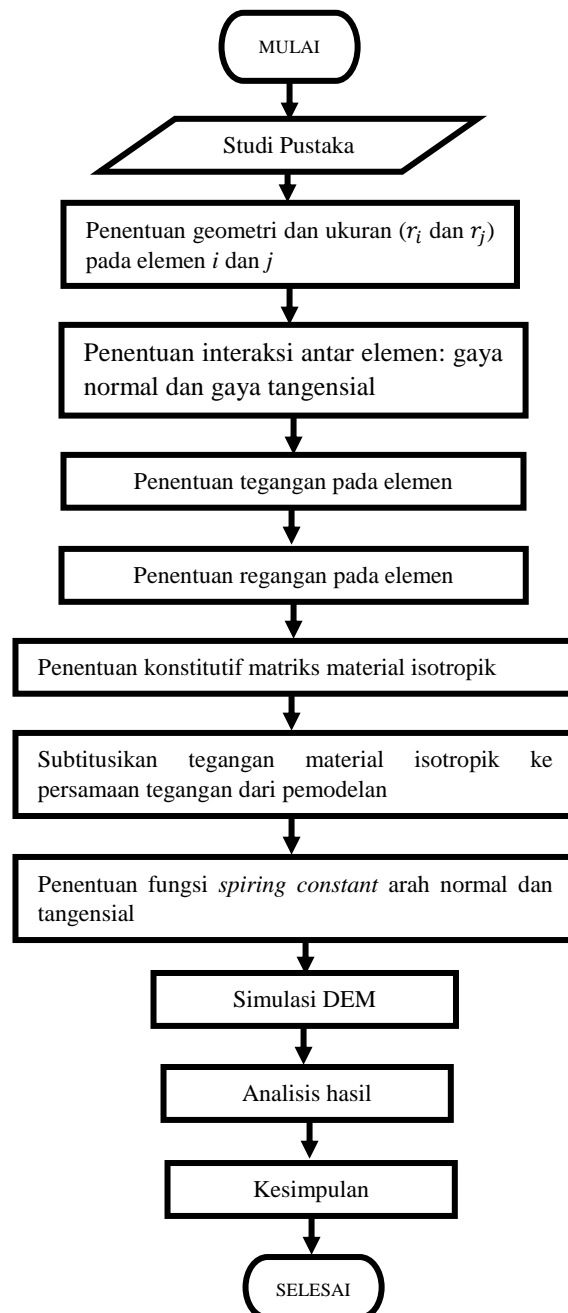
*(halaman ini sengaja dikosongkan)*

## BAB 3

### METODOLOGI PENELITIAN

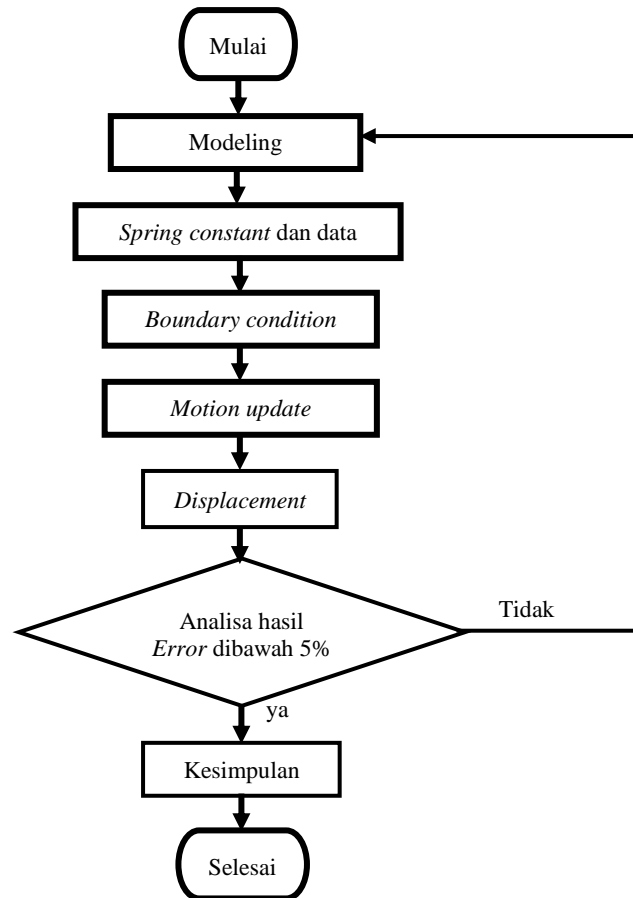
#### 3.1 Diagram Alir Penelitian

Langkah-langkah penelitian dilakukan dengan baik dan sistematis seperti yang terdapat pada Gambar 3.1 berikut.



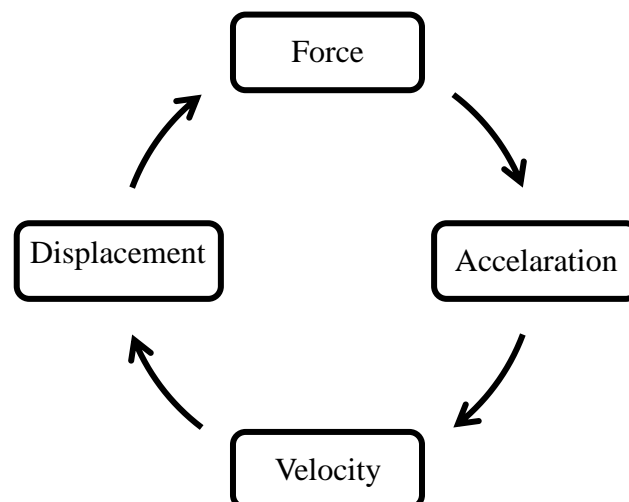
Gambar 3.1 Diagram Alir Metodologi Penelitian

Untuk validasi fungsi *spring constant* yang diperoleh, dilakukan simulasi. Adapun langkah-langkah simulasi dapat dilihat pada Gambar 3.2.



Gambar 3.2 Diagram Alir Simulasi DEM

*Displacement* pada simulasi DEM ditentukan dengan menentukan *force*, *acceleration*, dan *velocity*, seperti terlihat pada Gambar 3.3.



Gambar 3.3 DEM calculation cycle

### 3.2 Tahapan Penelitian

Secara umum tahapan penelitian yang dilakukan, antara lain sebagai berikut:

1. Studi Pustaka

Studi pustaka dilakukan dengan tujuan untuk menambah pemahaman terkait DEM. Studi pustaka dapat diperoleh dari penelitian-penelitian sebelumnya tentang DEM.

2. Formulasi  $k_n$  dan  $k_s$

Pada tahap ini akan dilakukan formulasi untuk memperoleh fungsi  $k_n$  dan  $k_s$ .

3. Simulasi DEM

Untuk mengetahui keakuratan dan efisiensi dari  $k_n$  dan  $k_s$  yang diperoleh, maka akan dilakukan simulasi. Simulasi dilakukan dengan *software matlab*.

4. Analisis hasil

Pada tahap ini, dilakukan analisis hasil yang diperoleh pada tahap simulasi. Selanjutnya akan dibandingkan dengan hasil penelitian sebelumnya.

5. Kesimpulan

### 3.3 Data

Untuk melakukan simulasi, diperlukan data-data properti material dan parameter DEM. Properti material dan parameter DEM ditunjukkan pada Tabel 3.1.

Tabel 3.1 Properti material dan Parameter DEM

Material	$E(GPa)$	$\nu$	$\rho \text{ ( kg/m}^3\text{)}$	NoE	$r_i(mm)$	$r_j(mm)$	$\Delta t \text{ (s)}$
Steel	210	0.27	7800	177	0.25	0.5	$1 \times 10^{-6}$

### 3.4 Formulasi Numerik

Langkah-langkah untuk memformulasikan *spring constant* arah normal ( $k_n$ ) dan *spring constant* arah tangensial ( $k_s$ ), adalah sebagai berikut:

1. Penentuan radius elemen  $i$  ( $r_i$ ) dan radius elemen  $j$  ( $r_j$ ) pada elemen.
2. Penentuan interaksi antar elemen: gaya normal ( $f_n$ ) dan gaya geser ( $f_s$ ).



Gaya normal ( $f_n$ ) dan gaya geser ( $f_s$ ) dapat ditentukan dengan persamaan berikut:

$$f_n = k_n \cdot \Delta u_n \quad (3.1)$$

$$f_s = k_s \cdot \Delta u_s \quad (3.2)$$

3. Penentuan tegangan pada elemen dengan persamaan:

$$\sigma' = \frac{f}{A} \quad (3.3)$$

dengan  $\sigma'$  adalah tegangan,  $f$  adalah gaya dan  $A$  adalah luas penampang.

4. Penentuan regangan yang terjadi pada elemen dengan persamaan:

$$\varepsilon = \Delta u / u \quad (3.4)$$

dengan  $\varepsilon$  adalah regangan,  $\Delta u$  adalah *displacement*, dan  $u$  adalah panjang awal.

5. Penentuan matriks konstitutif material isotropik. Adapun matriks konstitutif material isotropik adalah sebagai berikut:

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{bmatrix} = \underbrace{\begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}}_C \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix} \quad (3.5)$$

Dengan:

$C$  adalah matriks konstituif material isotropik

$$c_{11} = c_{22} = \frac{E}{(1 - \nu^2)} \quad (3.6)$$

$$c_{12} = c_{21} = \frac{E\nu}{(1 - \nu^2)} \quad (3.7)$$

$$c_{33} = \frac{E}{2(1 + \nu)} \quad (3.8)$$

6. Substitusi tegangan material isotropik (matriks konstitutif) ke persamaan tegangan yang diperoleh dari pemodelan, secara matematis dapat ditulis menjadi:

$$\sigma' = \sigma \quad (3.9)$$

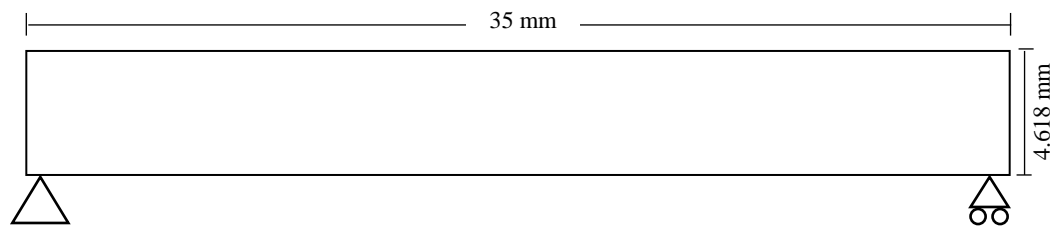
7. Penentuan fungsi  $k_n$  dan  $k_s$  sebagai fungsi dari modulus Young's ( $E$ ), Poisson ratio ( $\nu$ ), ukuran ( $r_i$  dan  $r_j$ ), dan ketebalan ( $\delta$ ).

### 3.5 Simulasi DEM

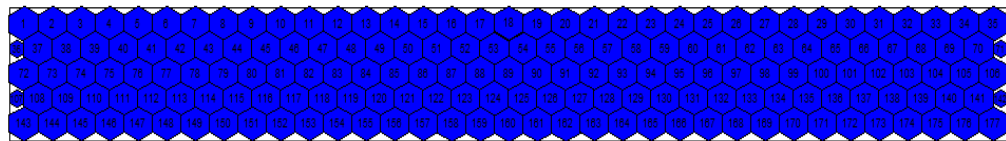
Untuk validasi dilakukan simulasi DEM. Adapun langkah-langkah simulasi adalah sebagai berikut:

#### 1. Modeling

Pada tahap ini dilakukan pemodelan susunan elemen pada simulasi seperti pada Gambar 3.5. Adapun dimensi material pada simulasi ini adalah ketebalan 1 mm, panjang 35 mm dan lebar 4.6188 mm seperti yang ditunjukkan pada Gambar 3.4.



Gambar 3.4 Dimensi material



Gambar 3.5 Susunan elemen pada simulasi

#### 2. Spring constant dan data

Pada tahap ini dilakukan penginputan fungsi *spring constant* arah normal ( $k_n$ ), *spring constant* arah tangensial ( $k_s$ ) yang telah diperoleh dari pemodelan dan penginputan data properti material serta parameter DEM.

#### 3. Boundary condition

Pada tahap ini dilakukan penentuan gaya dan *fix support*. Nilai gaya yang disimulasikan sebesar 100 N ke arah -y.

#### 4. Displacement

Gaya pada setiap elemen menentukan *displacement* yang terjadi pada elemen tersebut. Penentuan *displacement* dilakukan secara bertahap (*step-by-step*) dengan interval  $\Delta t$  (s). Konsep algoritmanya ditunjukkan pada Gambar 3.3. Penentuan nilai *displacement* dengan langkah-langkah sebagai berikut:

- a. Penentuan percepatan, kecepatan, delta x, y dan theta dengan persamaan sebagai berikut (liu,2004):

$$[\ddot{x}_i]_t = \frac{[Fx_i]_t}{m} \quad (3.10)$$

$$[\ddot{y}_i]_t = \frac{[Fy_i]_t}{m} \quad (3.11)$$

$$[\ddot{\phi}_i]_t = \frac{[M_i]_t}{I_i} \quad (3.12)$$

Setelah diperoleh percepatan, maka kecepatan, delta x, y dan theta dapat diperoleh dengan langkah sebagai berikut (liu, 2004):

$$[\dot{x}_i]_{t+\Delta t} = [\dot{x}_i]_{t-\Delta t} + \Delta t [\ddot{x}_i]_t \quad (3.13)$$

$$[x_i]_{t+\Delta t} = [x_i]_t + \Delta t [\dot{x}_i]_{t+\Delta t} \quad (3.14)$$

$$[\dot{x}_i]_t = \frac{1}{2} ([\dot{x}_i]_{t-\Delta t} + [\dot{x}_i]_{t+\Delta t}) \quad (3.15)$$

$$[\dot{y}_i]_{t+\Delta t} = [\dot{y}_i]_{t-\Delta t} + \Delta t [\ddot{y}_i]_t \quad (3.16)$$

$$[y_i]_{t+\Delta t} = [y_i]_t + \Delta t [\dot{y}_i]_{t+\Delta t} \quad (3.17)$$

$$[\dot{y}_i]_t = \frac{1}{2} ([\dot{y}_i]_{t-\Delta t} + [\dot{y}_i]_{t+\Delta t}) \quad (3.18)$$

$$[\dot{\phi}_i]_{t+\Delta t} = [\dot{\phi}_i]_{t-\Delta t} + \Delta t [\ddot{\phi}_i]_t \quad (3.19)$$

$$[\phi_i]_{t+\Delta t} = [\phi_i]_t + \Delta t [\dot{\phi}_i]_{t+\Delta t} \quad (3.20)$$

$$[\dot{\phi}_i]_t = \frac{1}{2} ([\dot{\phi}_i]_{t-\Delta t} + [\dot{\phi}_i]_{t+\Delta t}) \quad (3.21)$$

- b. Penentuan  $[X_i]_{t+\Delta t}$  dan  $[Y_i]_{t+\Delta t}$

$$[X_i]_{t+\Delta t} = [X_i]_t + \Delta x \quad (3.22)$$

$$[Y_i]_{t+\Delta t} = [Y_i]_t + \Delta y \quad (3.23)$$

- c. Penentuan  $\Delta u_n$  dan  $\Delta u_s$  dengan persamaan (liu, 2004):

$$\Delta u_n = -(\Delta x_i - \Delta x_j) \cos [\alpha_{ij}]_t - (\Delta y_i - \Delta y_j) \sin [\alpha_{ij}]_t \quad (3.24)$$

$$\Delta u_s = -(\Delta x_i - \Delta x_j) \sin [\alpha_{ij}]_t - (\Delta y_i - \Delta y_j) \cos [\alpha_{ij}]_t + (r_i \Delta \phi_i + r_j \Delta \phi_j) \quad (3.25)$$

- d. Penentuan  $[f_n]_t$  dan  $[f_s]_t$  dengan persamaan (liu, 2004):

$$[f_n]_t = [f_n]_{t-1} + k_n \cdot \Delta u_n \quad (3.26)$$

$$[f_s]_t = [f_s]_{t-1} + k_n \cdot \Delta u_n \quad (3.27)$$

e. Penentuan resultan gaya dan momen (liu, 2004):

$$[Fx_i]_t = \sum_j \left( [f_n^j]_t \cos [\alpha_{ij}^j]_t - [f_s^j]_t \sin [\alpha_{ij}^j]_t \right) \quad (3.28)$$

$$[Fy_i]_t = \sum_j \left( [f_n^j]_t \sin [\alpha_{ij}^j]_t + [f_s^j]_t \cos [\alpha_{ij}^j]_t \right) \quad (3.29)$$

$$[M_i]_t = -r_i \sum_j [f_s^j]_t \quad (3.30)$$

f. Kembali ke langkah pertama

### 3.6 Analisa Hasil

Pada tahap ini, dilakukan analisis hasil simulasi yang berupa grafik *displacement*. Hasil simulasi ini kemudian akan dibandingkan dengan hasil dari penelitian sebelumnya dan hitungan manual defleksi.

*(halaman ini sengaja dikosongkan)*

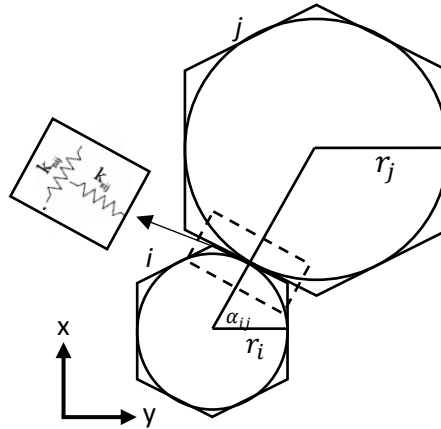
## BAB 4

### HASIL DAN PEMBAHASAN

#### 4.1. Formulasi *Spring Constant* Arah Normal dan Tangensial

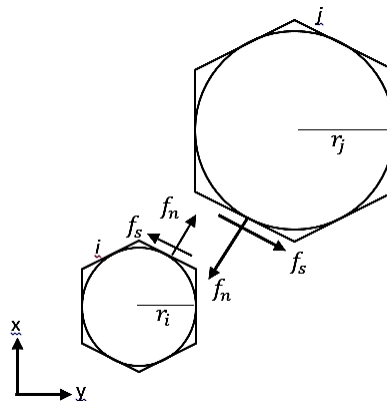
Langkah-langkah formulasi *spring constant* arah normal dan tangensial sebagai berikut.

1. Penentuan geometri untuk formulasi *spring constant* arah normal dan tangensial. Pada penelitian ini geometri yang digunakan adalah heksagonal yang menyinggung lingkaran berjari-jari  $r$  di dalamnya, sehingga ukuran dari elemen  $i$  dan  $j$  dapat dinilai dari jari-jarinya ( $r$ ). Geometri dan ukuran ( $r_i$  dan  $r_j$ ) pada elemen  $i$  dan  $j$ , seperti pada Gambar 4.1.



Gambar 4.1 Geometri dan Ukuran pada Elemen  $i$  dan  $j$

2. Penentuan interaksi antar elemen: gaya normal ( $f_n$ ) dan gaya geser ( $f_s$ ) pada elemen  $i$  dan  $j$  seperti yang ditunjukkan pada Gambar 4.2.



Gambar 4.2 Gaya pada Elemen  $i$  dan  $j$

Gaya normal ( $f_n$ ) dan gaya geser ( $f_s$ ) dapat ditentukan dengan pers. (4.1) dan (4.2).

$$f_n = k_n \cdot \Delta u_n \quad (4.1)$$

$$f_s = k_s \cdot \Delta u_s \quad (4.2)$$

dengan  $k_n$  adalah konstanta pegas normal,  $k_s$  adalah konstanta pegas geser,  $\Delta u_n$  adalah *displacement* normal dan  $\Delta u_s$  adalah *displacement* geser.

3. Penentuan regangan yang terjadi pada elemen dengan pers. (4.3) berikut:

$$\varepsilon = \Delta u / u \quad (4.3)$$

dengan  $\varepsilon$  adalah regangan,  $\Delta u$  adalah *displacement*, dan  $u$  adalah panjang awal. Regangan normal dan regangan geser pada pemodelan *spring constant* ini adalah sebagai berikut:

$$\varepsilon_{xx} = \varepsilon_{yy} = \varepsilon_n = \frac{\Delta u_n}{r_i + r_j} \quad (4.4)$$

$$\varepsilon_{xy} = \varepsilon_{yx} = \varepsilon_s = \frac{\Delta u_s}{l_i + l_j} = \frac{\Delta u_s \sqrt{3}}{2(r_i + r_j)} \quad (4.5)$$

$$\gamma_{xy} = \varepsilon_{xy} + \varepsilon_{yx} = \frac{\Delta u_s \sqrt{3}}{(r_i + r_j)} \quad (4.6)$$

penentuan luas penampang dengan pers. (4.7)

$$A = l \cdot \delta \quad (4.7)$$

dengan  $A$  adalah luas penampang,  $l$  adalah panjang sisi elemen, dan  $\delta$  adalah ketebalan elemen. Panjang sisi elemen heksagonal dengan 6 sisi yang menyinggung lingkaran berjari-jari  $r$  di dalamnya, dapat ditentukan dengan pers. (4.8) sebagai berikut:

$$\text{keliling} = 2nr \tan \frac{\pi}{n} = 2nr \tan \frac{180}{n} \quad (4.8)$$

dengan  $n$  adalah jumlah sisi, karena heksagonal mempunyai 6 sisi, maka pers. (4.8) dapat ditulis kembali menjadi:

$$\begin{aligned} \text{keliling} &= 2 \cdot 6 \cdot r \tan \frac{180}{6} \\ &= 12 \cdot r \cdot \tan 30 \\ &= 12 \cdot r \cdot \frac{1}{\sqrt{3}} = 4\sqrt{3} r \end{aligned}$$

Panjang sisi diperoleh dengan membagi nilai keliling dengan jumlah sisi, maka diperoleh:

$$l = \frac{4\sqrt{3} r}{6} = \frac{2\sqrt{3} r}{3}$$

Jadi luas penampangnya adalah:

$$A = (2\sqrt{3}r/3) \cdot \delta$$

4. Penentuan tegangan pada elemen dengan pers. (4.9):

$$\sigma' = \frac{f}{A} \quad (4.9)$$

dengan  $\sigma'$  adalah tegangan,  $f$  adalah gaya dan  $A$  adalah luas penampang.

Tegangan dibagi menjadi 2 yaitu tegangan normal dan tegangan geser. Pada formulasi ini diperoleh tegangan normal dan tegangan geser sebagai berikut:

$$\sigma'_{xx} = \frac{f_{nij}}{A} = \frac{\Delta u_n \cdot k_n}{\delta \cdot l} = \frac{\Delta u_n \cdot k_n \cdot \sqrt{3}}{\delta \cdot 2 r_i} \quad (4.10)$$

$$\sigma'_{yy} = \frac{f_{nij}}{A} = \frac{\Delta u_n \cdot k_n}{\delta \cdot l} = \frac{\Delta u_n \cdot k_n \cdot \sqrt{3}}{\delta \cdot 2 r_i} \quad (4.11)$$

$$\sigma'_{xy} = \sigma_{yx} = \frac{f_s}{A} = \frac{\Delta u_s \cdot k_s}{\delta \cdot l} = \frac{\Delta u_s \cdot k_s \cdot \sqrt{3}}{\delta \cdot 2 r_i} \quad (4.12)$$

$$\tau'_{xy} = \frac{1}{2}(\sigma_{xy} + \sigma_{yx}) = \frac{\Delta u_s \cdot k_s \cdot \sqrt{3}}{\delta \cdot 2 r_i} \quad (4.13)$$

Substitusikan pers. (4.4) ke pers. (4.10), (4.11) dan pers. (4.5) ke pers. (4.13) sehingga diperoleh:

$$\sigma'_{xx} = \frac{k_n \cdot \sqrt{3} \cdot (r_i + r_j)}{\delta \cdot 2 r_i} \varepsilon_{xx} \quad (4.14)$$

$$\sigma'_{yy} = \frac{k_n \cdot \sqrt{3} \cdot (r_i + r_j)}{\delta \cdot 2 r_i} \varepsilon_{yy} \quad (4.15)$$

$$\tau'_{xy} = \frac{k_s \cdot (r_i + r_j)}{\delta \cdot 2 r_i} \gamma_{xy} \quad (4.16)$$



Jika pers. (4.14), (4.15) dan (4.16) dituliskan dalam bentuk matriks menjadi pers. (4.17).

$$\begin{bmatrix} \sigma'_{xx} \\ \sigma'_{yy} \\ \tau'_{xy} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{k_n \cdot \sqrt{3} \cdot (r_i + r_j)}{\delta \cdot 2 r_i} & 0 & 0 \\ 0 & \frac{k_n \cdot \sqrt{3} \cdot (r_i + r_j)}{\delta \cdot 2 r_i} & 0 \\ 0 & 0 & \frac{k_s \cdot (r_i + r_j)}{\delta \cdot 2 r_i} \end{bmatrix}}_{C'} \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix} \quad (4.17)$$

##### 5. Penentuan matriks konstitutif isotropik

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{bmatrix} = \underbrace{\begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}}_C \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix} \quad (4.18)$$

dimana C adalah matriks konstitutif material isotropik

$$c_{11} = c_{22} = \frac{E}{(1 - \nu^2)} \quad (4.19)$$

$$c_{12} = c_{21} = \frac{E\nu}{(1 - \nu^2)} \quad (4.20)$$

$$c_{33} = \frac{E}{2(1 + \nu)} \quad (4.21)$$

Untuk menjaga ekuivalen antara C dan C', maka digunakan *eigenvalue*. Cara penentuan *eigenvalue* dengan menggunakan pers. (4.22) berikut:

$$\det(C - \lambda I) \quad (4.22)$$

jika dituliskan kembali:

$$\begin{aligned} \det(C - \lambda I) &= \begin{vmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{vmatrix} - \lambda \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} \\ &= \begin{vmatrix} c_{11} - \lambda & c_{12} & c_{13} \\ c_{21} & c_{22} - \lambda & c_{23} \\ c_{31} & c_{32} & c_{33} - \lambda \end{vmatrix} \\ &= (c_{11} - \lambda)(c_{22} - \lambda)(c_{33} - \lambda) - c_{12} \cdot c_{21} (c_{33} - \lambda) \end{aligned} \quad (4.23)$$

Karena  $c_{11} = c_{22}$  dan  $c_{12} = c_{21}$ , maka persamaan (4.23) dapat ditulis kembali:

$$\det(C - \lambda I) = (c_{11} - \lambda)(c_{11} - \lambda)(c_{33} - \lambda) - c_{12} \cdot c_{12} (c_{33} - \lambda) = 0$$

Sehingga diperoleh:

$$\lambda_1 = c'_{11} = c_{11} + c_{12} \quad (4.24)$$

$$\lambda_2 = c'_{22} = c_{11} - c_{12} \quad (4.25)$$

$$\lambda_3 = c'_{33} = c_{33} \quad (4.26)$$

6. Penentuan *spring constant* arah normal ( $k_n$ ) dan *spring constant* arah tangensial ( $k_s$ ) sebagai berikut.

▪ *spring constant* arah normal ( $k_n$ )

$$\begin{aligned} c'_{11} &= c'_{11} \\ \frac{k_n \cdot \sqrt{3} \cdot (r_i + r_j)}{\delta \cdot 2 r_i} &= c_{11} + c_{12} \\ k_n &= \frac{\delta \cdot 2\sqrt{3} r_i}{3 (r_i + r_j)} (c_{11} + c_{12}) \\ &= \frac{\delta \cdot 2\sqrt{3} r_i}{3 (r_i + r_j)} \left( \frac{E}{(1 - v^2)} + \frac{Ev}{(1 - v^2)} \right) \\ &= \frac{\delta \cdot 2\sqrt{3} r_i}{3 (r_i + r_j)} \cdot \frac{E}{1 - v} \end{aligned}$$

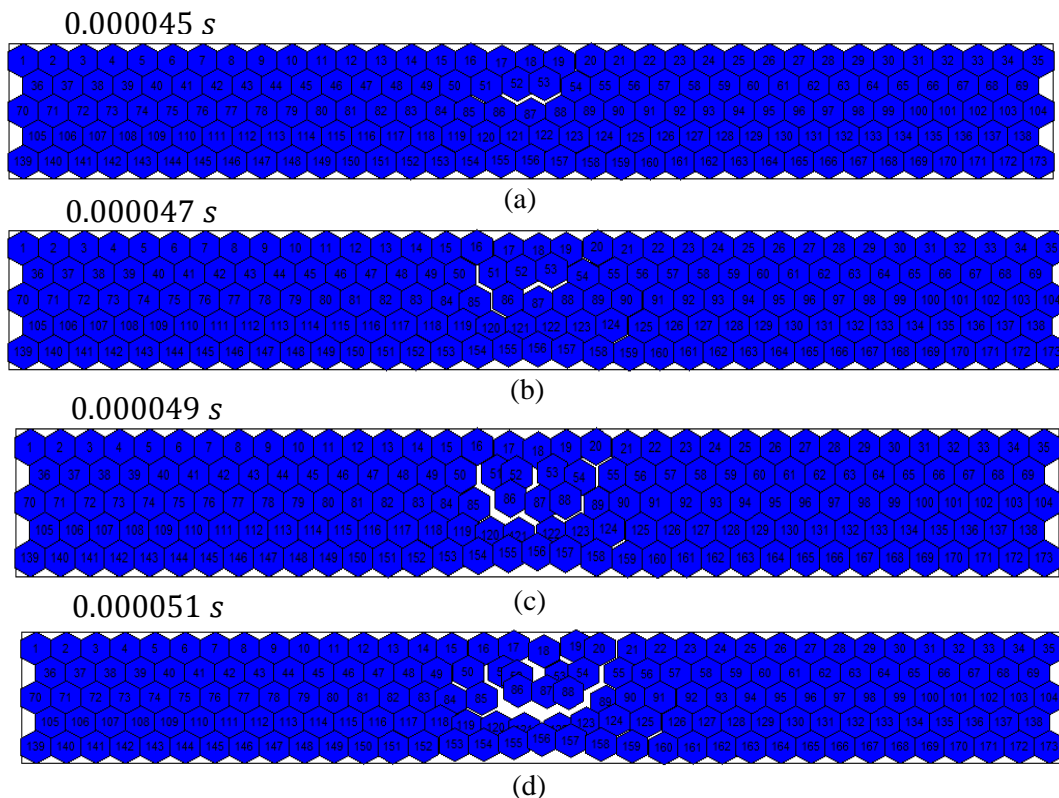
▪ *spring constant* arah tangensial ( $k_s$ )

$$\begin{aligned} c'_{33} &= c'_{33} \\ \frac{k_s \cdot (r_i + r_j)}{\delta \cdot 2 r_i} &= c_{33} \\ k_s &= \frac{\delta \cdot 2 r_i}{r_i + r_j} \cdot c_{33} \\ &= \frac{\delta \cdot r_i}{r_i + r_j} \cdot \frac{E}{1 + v} \end{aligned}$$

dengan  $E$  adalah modulus young,  $v$  adalah poisson rasio,  $\delta$  adalah ketebalan, dan  $r_i$  dan  $r_j$  adalah jari-jari elemen  $i$  dan  $j$ .

#### 4.2. Hasil Simulasi dengan Menggunakan *Spring Constant* yang Diperoleh Gaeni

Validasi *spring constant* yang diperoleh dari formulasi dilakukan dengan cara simulasi. Validasi yang menjadi acuan yaitu fungsi *spring constant* yang diperoleh oleh Gaeni (2008), fungsi yang diperoleh dengan pendekatan dua elemen yang memiliki ukuran yang sama (pers. 2.24 dan 2.25). Properties material dan parameter DEM yang digunakan dalam simulai sama seperti yang ditunjukkan pada Tabel 3.1, namun jumlah elemen yang disimulasikan sebanyak 173 elemen. Parameter berupa dimensi spesimen memiliki nilai yang sama (Gambar 3.4). Tujuan simulasi ini untuk membandingkan hasil simulasi yang diperoleh Gaeni (2008) dan hasil simulasi menggunakan *spring constant* yang diperoleh dari formulasi. Simulasi ditunjukkan pada Gambar 4.3 sebagai berikut.

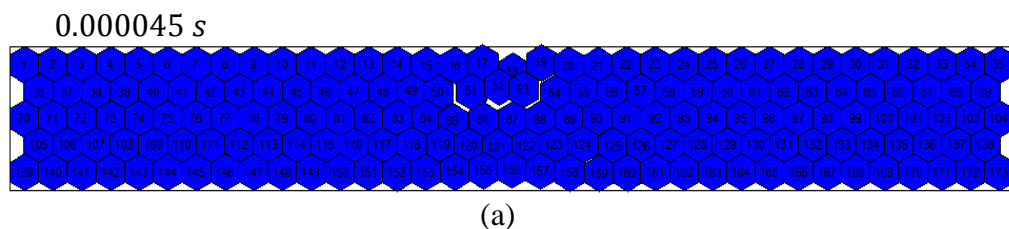


Gambar 4.3 Simulasi dengan *Spring Constant* yang Diperoleh Gaeni dengan Jumlah Elemen 173

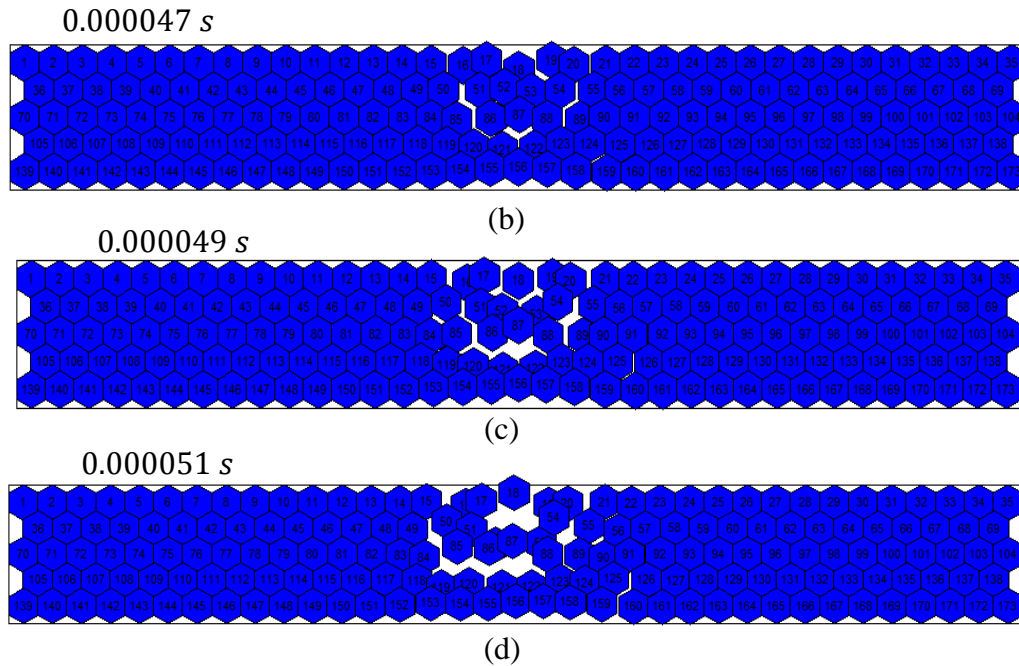
Gambar 4.3 merupakan hasil simulasi menggunakan *spring constant* yang diperoleh Gaeni pada saat 0.000045 s sampai 0.000051 s. Dari Gambar 4.3 dapat diketahui bahwa pada saat 0.000045 s mulai terjadinya *displacement* pada elemen. Pada simulasi tersebut, yang menjadi fokus pengamatan adalah elemen ke 87. Dari hasil simulasi dapat diketahui bahwa nilai *displacement* yang terjadi pada Gambar 4.3a diperoleh sebesar -0.0206 mm. Pada saat 0.000047 s, maka *displacement* yang terjadi meningkat menjadi 0.0195 mm. Dengan demikian semakin bertambah waktu, menyebabkan *displacement* yang terjadi semakin besar. Hal ini dapat dilihat pada Gambar 4.3c dan 4.3d dengan nilai *displacement* yang terjadi masing-masing sebesar 0.0705 dan 0.135.

#### 4.3. Hasil Simulasi dengan Menggunakan *Spring Constant* yang Diperoleh dari Pemodelan

Hasil simulasi yang ditunjukkan pada Gambar 4.4 diperoleh dengan menggunakan *spring constant* dari formulasi (pers. 4.27 dan 4.38). Dari simulasi ini, akan diamati perubahan *displacement* ke arah y yang terjadi pada elemen ke 87. Gambar 4.4a merupakan hasil simulasi pada saat 0.000045 s, nilai *displacement* yang diperoleh sebesar 0.0312 mm. Pada saat 0.000047 s, *displacement* yang terjadi ditunjukkan pada Gambar 4.4b dan 4.4c dengan nilai masing-masing sebesar 0.077 mm dan 0.1023 mm, sedangkan nilai *displacement* pada Gambar 4.4d sebesar 0.0624 mm.



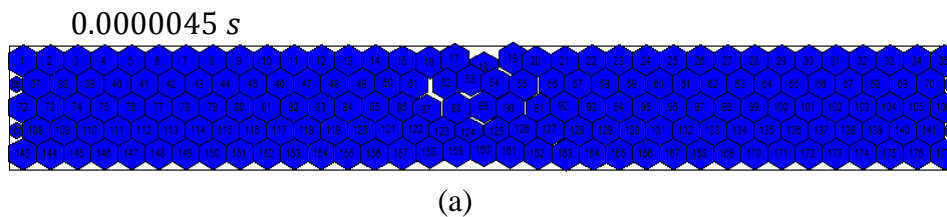
Gambar 4.4 Simulasi dengan *Spring Constant* yang Diperoleh pada Penelitian Ini dengan Jumlah Elemen 173



Gambar 4.4 Simulasi dengan *Spring Constant* yang Diperoleh pada Penelitian Ini dengan Jumlah Elemen 173 lanjutan

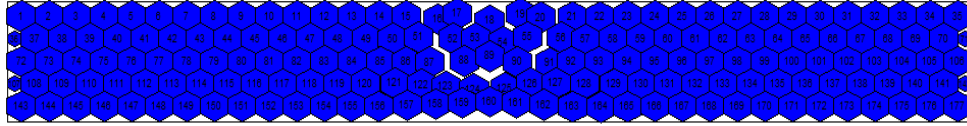
#### 4.4. Hasil Simulasi dengan Menggunakan *Spring Constant* yang Diperoleh dari Pemodelan dengan Jumlah 177 Elemen

Hasil simulasi yang ditunjukkan pada Gambar 4.5 diperoleh dengan menggunakan *spring constant* yang diperoleh dari formulasi (pers. 4.27 dan 4.38). Dari simulasi ini diamati perubahan *displacement* pada arah y yang terjadi pada elemen ke 89. Gambar 4.5a merupakan hasil simulasi pada saat 0.000045 s, nilai *displacement* yang diperoleh sebesar -0.0150 mm. Pada saat 0.000047 s dan 0.000049, *displacement* yang terjadi ditunjukkan pada Gambar 4.5b dan 4.5c dengan nilai masing-masing sebesar 0.0142 mm dan 0.0728 mm, sedangkan nilai *displacement* pada Gambar 4.5d sebesar 0.1346 mm.



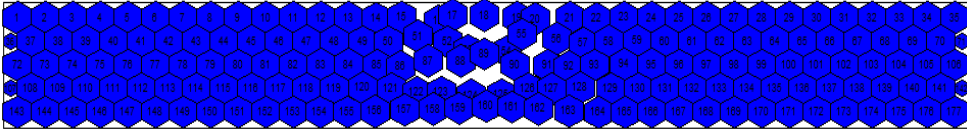
Gambar 4.5 Simulasi dengan *Spring Constant* yang Diperoleh pada Penelitian Ini dengan Jumlah Elemen 177 Lanjutan

0.0000047 s



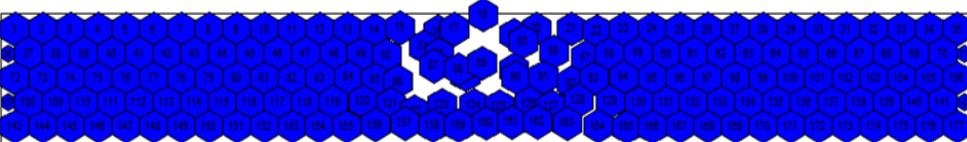
(b)

0.0000049 s



(c)

0.0000051 s

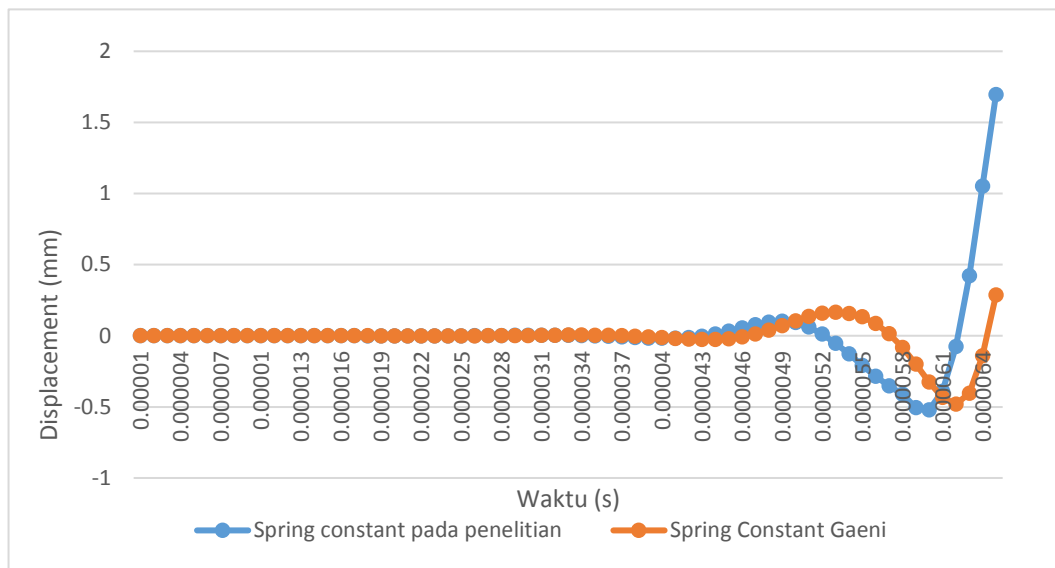


(d)

Gambar 4.5 Simulasi dengan *Spring Constant* yang Diperoleh pada Penelitian Ini dengan Jumlah Elemen 177 Lanjutan

#### 4.5. Hasil Perbandingan *Displacement* Antara Simulasi dengan *Spring Constant* Gaeni dan *Spring Constant* dari Formulasi

*Displacement* didefinisikan sebagai nilai dari perubahan posisi benda dari posisi awal ke posisi akhir yang dipengaruhi oleh kecepatan dan waktu. Pada hasil simulasi yang ditunjukkan pada Gambar 4.6 merupakan grafik yang dihasilkan dari perbandingan simulasi menggunakan *spring constant* dari formulasi dan simulasi menggunakan *spring constant* yang diperoleh Gaeni.



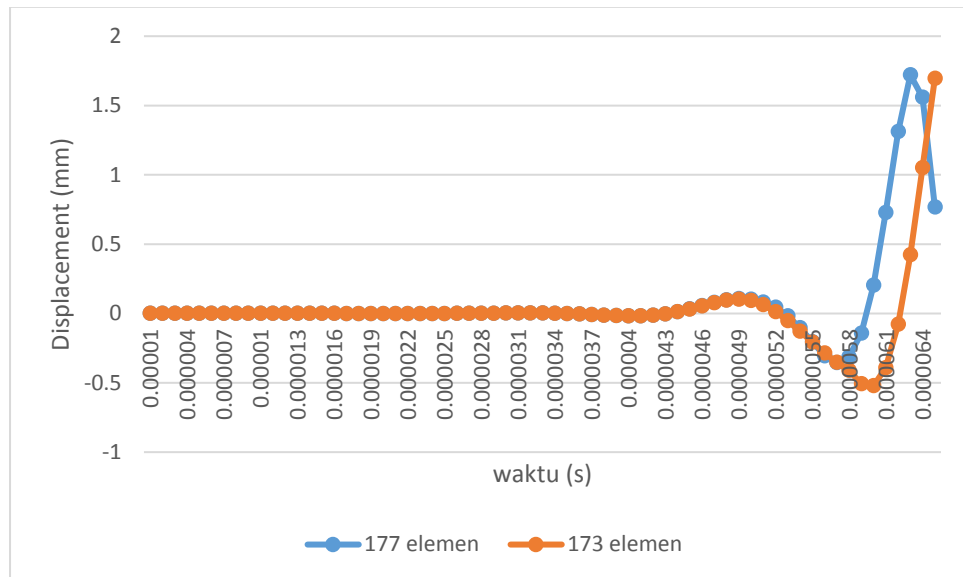
Gambar 4.6 Grafik Perbandingan Simulasi Menggunakan *Spring Constant* yang Diperoleh dari Formulasi dan *Spring Constant* dari Formulasi Gaeni

Hasil grafik perbandingan menunjukkan pola dan *tren line* yang sama pada waktu  $1 \times 10^{-5} s$  sampai  $43 \times 10^{-4} s$ . Dengan demikian dapat dikatakan bahwa gaya yang diberikan pada spesimen belum memberi efek pada spesimen tersebut. Namun ketika waktu bertambah menjadi lebih dari  $43 \times 10^{-4} s$ , *displacement* yang terjadi mulai terlihat perbedaannya. Hal tersebut terjadi sampai waktu akhir  $65 \times 10^{-4} s$ . *Displacement* maksimum yang terjadi pada waktu  $65 \times 10^{-4} s$  adalah  $-0.5217 mm$ . Jika nilai *displacement* yang diperoleh dari simulasi *spring constant* dari formulasi dibandingkan dengan simulasi *spring constant* Gaeni, dihasilkan nilai yang hampir sama yaitu *displacement* minimum sebesar  $-0.48103$ .

Oleh karena itu, sebagai kesimpulan bahwa nilai *displacement* yang terjadi dipengaruhi oleh gaya, properti material steel berupa densitas, modulus young, poisson rasio, dan parameter DEM. Ketika gaya, properti dan parameter DEM berbeda, maka *displacement* yang terjadi berbeda.

#### 4.6. Hasil Analisis *Displacement* pada Simulasi dengan Jumlah Elemen yang Berbeda

Perbandingan *displacement* pada simulasi dengan *spring constant* yang diperoleh dari formulasi, namun jumlah elemen berbeda ditunjukkan pada Gambar 4.7.



Gambar 4.7 Grafik Perbandingan *Displacement* Kearah y dengan Jumlah Elemen 173 dan 177

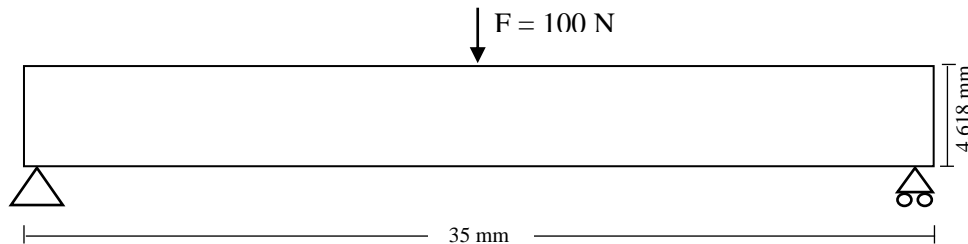
Gambar 4.7 menunjukan tren grafik memiliki arah yang hampir sama dari 0 s sampai  $4 \times 10^{-5}$  s. Pada saat  $4 \times 10^{-5}$  s nilai *displacement* dari simulasi dengan jumlah elemen sebanyak 177 dan 173 masing-masing sebesar -0.0181 mm dan -0.08122 mm. Sedangkan semakin bertambah waktunya, maka *displacement* yang terjadi mulai terlihat perbedaannya. Hal ini dapat diketahui dari grafik yang menunjukkan penurunan. Namun penurunan arah radial (-y) sampai nilai *displacement* minimum, terjadi pada waktu yang berbeda. Pada simulasi dengan 177 elemen, *displacement* minimum terjadi pada waktu  $5.8 \times 10^{-5}$  s dengan nilai sebesar -0.31342 mm. Selanjutnya, arah perubahan *displacement* mulai terjadi ketika waktunya semakin meningkat. Hal ini ditunjukkan pada Gambar 4.7 bahwa perubahan *displacement* maksimum arah radial (+y) sebesar 1.720145 mm pada waktu  $6.3 \times 10^{-5}$  s.



Pada simulasi dengan 173 elemen, nilai *displacement* minimum terjadi pada arah radial (-y) sebesar -0.52147 mm, hal ini terjadi pada waktu  $6 \times 10^{-5}$  s. Selanjutnya, pada saat waktu bertambah, maka mengakibatkan *displacement* arah radial (+y) maksimal yang terjadi pada waktu  $6.5 \times 10^{-5}$  s sebesar 1.69003 mm.

#### 4.7. Perbandingan Hasil Analisis Defleksi Spesimen

Defleksi adalah perubahan bentuk yang terjadi pada balok ke arah *radial* (-y) akibat adanya pemberian gaya vertikal pada balok. Defleksi diukur dari permukaan netral awal ke posisi netral akhir. Hasil perhitungan defleksi dari spesimen dengan ukuran lebar 4.618 mm dan ketebalan 1 mm pada gaya (F) 100 N (Gambar 4.8). Untuk mendapatkan nilai defleksi dari spesimen yang digunakan, terlebih dahulu menghitung inersia pada beam. Nilai inersia yang diperoleh dari perhitungan sebesar  $1.778 \text{ mm}^4$ . Sehingga nilai defleksi yang diperoleh dari perhitungan sebesar -0.0005856 mm. Nilai hasil perhitungan defleksi akan dibandingkan dengan hasil simulasi.



Gambar 4.8 Dimensi dan Pemberian Gaya

Metode analitis yang digunakan menggunakan persamaan berikut:

Nilai inersia dihitung dengan pers. (4.27) berikut:

$$I = \frac{b h^3}{12} \quad (4.27)$$

dengan  $b$  adalah ketebalan dan  $h$  adalah lebar.

Nilai inersia yang terjadi adalah:

$$I = \frac{b h^3}{12} = \frac{1 \text{ mm} \times (4.1688 \text{ mm})^3}{12} = 1.778 \text{ mm}^4$$

$$y_c = \frac{F \cdot l^3}{48 E I} \quad (4.28)$$

dengan  $F$  adalah gaya,  $l$  adalah panjang,  $E$  adalah modulus young dan  $I$  adalah inersia.

Defleksi yang terjadi adalah:

$$y_c = \frac{F \cdot l^3}{48 E I} = \frac{-100 \text{ N} \times (35 \text{ mm})^3}{48 \times (210 \times 10^3 \text{ N/mm}^2) \times 1.778 \text{ mm}^4} = -0.000585 \text{ mm}$$

Hasil perbandingan analisis defleksi antara hasil yang diperoleh dari perhitungan analitis, simulasi menggunakan *spring constant* yang diperoleh Gaeni dengan elemen sebanyak 173, dan simulasi dengan *spring constant* yang diperoleh dari formulasi dengan elemen sebanyak 173 dan 177. Hasil perbandingannya berupa nilai *error* yang ditunjukkan pada Tabel 4.1.

Tabel 4.1 Perhitungan *error* dari perbandingan perhitungan analitik dan simulasi

No	Metode	Jumlah elemen	Posisi elemen yang diamati	Time (s)	Displacement (mm)	Error (%)
1	Simulasi dengan spring constant Gaeni	173	87	0.00002	-0.000499	14.76119
2	Simulasi dengan spring constant dari formulasi	173	87	0.00002	-0.000557	4.88332
3	Simulasi dengan spring constant dari formulasi	177	89	0.00002	-0.000556	4.89486
4	Analitik	1			-0.000585	

Nilai yang ditunjukkan pada Tabel 4.1 diketahui bahwa *displacement* yang diperoleh dari simulasi dengan *spring constant* Gaeni adalah -0.000499 mm dan apabila dibandingkan dengan hasil perhitungan analitis defleksi diperoleh nilai *error* sebesar 14.76960 %. Sedangkan nilai *displacement* dari simulasi dengan elemen sebanyak 173 dan 177 elemen menggunakan *spring constant* yang diperoleh dari formulasi diperoleh masing-masing sebesar -0.000557 mm dan -0.000556 mm. Apabila kedua nilai *displacement* tersebut dibandingkan dengan nilai perhitungan analitis defleksi, diperoleh nilai *error* masing-masing sebesar 4.88332 % dan 4.89862 %.

*(halaman ini sengaja dikosongkan)*

## BAB 5

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, dari tahap formulasi dan simulasi dengan *discrete element method*, maka dapat diperoleh kesimpulan sebagai berikut:

1. Hasil analitik terkait dengan *Spring constant* arah normal ( $k_n$ ) dan *spring constant* arah tangensial ( $k_s$ ) diperoleh persamaan

$$k_n = \frac{\delta \cdot 2\sqrt{3} r_i}{3 (r_i + r_j)} \cdot \frac{E}{1 - \nu}$$
$$k_s = \frac{\delta \cdot r_i}{r_i + r_j} \cdot \frac{E}{1 + \nu}$$

dengan  $k_n$  dan  $k_s$  adalah *spring constant* arah normal dan tangensial,  $E$  adalah modulus Young's,  $\nu$  adalah poisson ratio,  $r_i$  dan  $r_j$  adalah radius dari elemen  $i$  dan  $j$ , dan  $\delta$  adalah ketebalan.

2. Hasil simulasi diperoleh nilai *displacement* sebesar -0.000556 mm, sedangkan hasil analitik diperoleh sebesar -0.000585 mm. Perbandingan nilai *error* dari dua analisa tersebut diperoleh sebesar 4.89486%. Dengan demikian, nilai *error* dari spring konstan pada studi ini, relatif lebih rendah dibanding hasil penelitian sebelumnya.

#### 5.2 Saran

Formulasi *spring constant* yang diperoleh dari hasil studi ini, dapat dikembangkan untuk aplikasi yang berbeda seperti simulasi yang berkaitan masalah *impact*.

*(halaman ini sengaja dikosongkan)*

## DAFTAR PUSTAKA

- Cundal, P.A. dan Strack. (1979), "A Discrete Numerical Model for Granular Assemblies", *Geotechnique*, Vol.29, hal.47–65.
- Dieter, G.E, (1988), *Mechanical Metallurgy*, 3<sup>rd</sup> edition, McGraw-Hill, Inc., New York.
- Gaeni, M., Mihardi S. dan Homma H. (2008), "Discrete Element Model for Continuum Dynamic Problems, *Journal of Solid Mechanics and Material Engineering*", Vol. 2, No. 12, hal. 1478-1486.
- Griffiths, D. V. dan Mustoe, G. G. W. (2001), "Modelling of Elastic Continua Using Grillage of Structural Element Based on Discrete Element Concepts", *Int. J. Numer. Meth. Engng*, Vol. 50, hal. 1759-1775.
- Liu, K., Gao, L. dan Tanimura, S. (2004), "Application of Discrete Element Method in Impact Problems", *JSME International Journal*, Series A, Vol. 47, No. 2, hal. 138-145.
- Pined, M. D, (2010), *Mekanika Kekuatan Material Lanjut*, edisi satu, Rekayasa Sains, Bandung.
- Prasetyo, Anton. (2009), "Kajian Shear Strain".
- Purwantana, Bambang. (2004), "Distinct Element Method dan Perspektif Penerapannya dalam Keteknikan Pertanian".
- Sawamoto, Y., Tsubota, H., Kasai, Y., Koshika, N. dan Morikawa, H. (1998), "Analytical Studies on Local Damage to Reinforced Concrete Structures under Impact Loading by Discrete Element Method", *Nuclear Engineering and Design*, Vol.179, hal.157–177.
- Sutikno, Homma H. dan Mihardi S. (2009), "Analysis on Contact Phenomenon under Particle Impact by Hybrid Method, *Journal of Solid Mechanics and material Engineering*", Vol. 3, No. 8, hal. 1010-1021.
- Sutikno, Homma H. dan Mihardi S. (2010), "Effect of Particle Fragmentation on the Stress Evolution in the Beam Impacted by a particle using Hybrid Method, *Journal of Solid Mechanics and material Engineering*", Vol. 4, No. 4, hal. 508-519.

Tavarez, A.F, dan Plesha, M.E. (2007), “Discrete Element Method for Modeling Solid and Particulate Materials”, *Int. Numer. Meth. Eng.*, Vol. 70, hal. 379-404.

## Lampiran 1. Coding Simulasi DEM

### *Discrete Element Method* (simulasi DEM)

```
%% Starting Program | Cleaning Workspace
clear all
clc
clf

%% Initialization Simulation

% Time Variable
t=0.0; % start time of simulation
tend=.0002; % end time of simulation
dt=0.000001; % time between each step
nt=round((tend-t)/(dt)); % number of steps

% Material Properties
E=21.0e+4; % Modulus Young's [newton/mm^2]
nu=0.3; % poisson's ratio
del=1; % thickness [mm]
rho=7.85e-6; % density of particle [kg/mm^3]

% Geometry
calc_mode=2; % Choose Mode
l=100; % length of beam [mm]
h=30; % height of beam [mm]
r1=0.25; % radius of element 1 [mm]
r2=0.5; % radius of element 2 [mm]
g=9.8;
n_col=35; % number of column
n_row=5; % number of row

% Calculate Distance Between Each Particle

a1=r1+r2; % horizontal1
b1=(2/3)*(3^(0.5))*r1; % vertical1
a2=2*r2; % horizontal2
b2=(2/3)*(3^(0.5))*r2; % vertical2

% Calculate Column & Row Configuration
if calc_mode==1
    [n_col,n_row]=element_1(l,h,a2,b2);
else
    if calc_mode==2
        %Calculate Length & Width of Beam
        [l,h]=element_2(n_col,n_row,a2,b2);
    else
        error('Unknown Mode !!!','File Error');
        clear
    end
end
```



```

end
end

% Calculate Number of Element
n_element = (n_row*n_col+(floor(n_row/2)));

%% Init Simulation Initial Condition
%Preallocation Memory

%Position
x=zeros(n_element,nt+6);
y=zeros(n_element,nt+6);
tetha=zeros(n_element,nt+6);

%Velocity
x_dot=zeros(n_element,nt+6);
y_dot=zeros(n_element,nt+6);
tetha_dot=zeros(n_element,nt+6);

%Acceleration
x_ddot=zeros(n_element,nt+6);
y_ddot=zeros(n_element,nt+6);
tetha_ddot=zeros(n_element,nt+6);

%Independent Displacement
U=zeros(n_element,nt+6);
V=zeros(n_element,nt+6);
TETHA=zeros(n_element,nt+6);

%Interaction Based Displacement
un=zeros(n_element,6,nt+6);
us=zeros(n_element,6,nt+6);

%Contact Force between Two Particle
fn=zeros(n_element,6,nt+6);
fs=zeros(n_element,6,nt+6);

%Contact Force with Each Neighborhood
FN=zeros(n_element,6,nt+6);
FS=zeros(n_element,6,nt+6);

%Force Component
fX=zeros(n_element,6,nt+6);
fY=zeros(n_element,6,nt+6);

%Angle between Two Particle
r_ij=zeros(n_element,6,nt+6);
sin=zeros(n_element,6,nt+6);
cos=zeros(n_element,6,nt+6);

% Calculate Stiffness Contact, Particle Mass, and Particle Inertia

```

```

[kn1,ks1,kn2,ks2,m1,m2,I1,I2]=element_Prop(E,nu,del,r1,r2,rho);

% Calculate Initial Position of Each Particle
[X,Y] = InitPosition01(n_element,h,r1,r2,n_col,n_row,a2,b2,nt);

% Calculate Neighbor Member
[T]=Neighborhood(n_element,X,Y,r2,nt);

% Init Applied Force to Specific Particle
fy=zeros(n_element,nt+6);
fx=zeros(n_element,nt+6);

fy(18,:)= -100;

% Total Force Applied to Each Particle
FX=zeros(n_element,nt);
FY=zeros(n_element,nt);
M0=zeros(n_element,nt);

%%
=====
= %%
%Simulation Process

for it=1:nt

    for i=1:n_element
        if i==36
            x_ddot(i,it)=(FX(i,it)+fx(i,it))/m1;
            y_ddot(i,it)=(FY(i,it)+fy(i,it))/m1;
            tetha_ddot(i,it)=M0(i,it)/I1;
        elseif i==71
            x_ddot(i,it)=(FX(i,it)+fx(i,it))/m1;
            y_ddot(i,it)=(FY(i,it)+fy(i,it))/m1;
            tetha_ddot(i,it)=M0(i,it)/I1;
        elseif i==107
            x_ddot(i,it)=(FX(i,it)+fx(i,it))/m1;
            y_ddot(i,it)=(FY(i,it)+fy(i,it))/m1;
            tetha_ddot(i,it)=M0(i,it)/I1;
        elseif i==142
            x_ddot(i,it)=(FX(i,it)+fx(i,it))/m1;
            y_ddot(i,it)=(FY(i,it)+fy(i,it))/m1;
            tetha_ddot(i,it)=M0(i,it)/I1;
        else
            x_ddot(i,it)=(FX(i,it)+fx(i,it))/m2;
            y_ddot(i,it)=(FY(i,it)+fy(i,it))/m2;
            tetha_ddot(i,it)=M0(i,it)/I2;
        end

    if it==1

```

```

    %x calculation at first iteration
    x_dot(i,it+1)=0+dt*x_ddot(i,it);
    x(i,it+1)=x(i,it)+dt*x_dot(i,it+1);
    x_dot(i,it)=0.5*(0+x_dot(i,it+1));

    %y calculation at first iteration
    y_dot(i,it+1)=0+dt*y_ddot(i,it);
    y(i,it+1)=y(i,it)+dt*y_dot(i,it+1);
    y_dot(i,it)=0.5*(0+y_dot(i,it+1));

    %tetha calculation at first iteration
    tetha_dot(i,it+1)=0+dt*tetha_ddot(i,it);
    tetha(i,it+1)=tetha(i,it)+dt*tetha_dot(i,it+1);
    tetha_dot(i,it)=0.5*(0+tetha_dot(i,it+1));

else

    %x calculation
    x_dot(i,it+1)=x_dot(i,it-1)+dt*x_ddot(i,it);
    x(i,it+1)=x(i,it)+dt*x_dot(i,it+1);
    x_dot(i,it)=0.5*(x_dot(i,it-1)+x_dot(i,it+1));

    %y calculation
    y_dot(i,it+1)=y_dot(i,it-1)+dt*y_ddot(i,it);
    y(i,it+1)=y(i,it)+dt*y_dot(i,it+1);
    y_dot(i,it)=0.5*(y_dot(i,it-1)+y_dot(i,it+1));

    %tetha calculation
    tetha_dot(i,it+1)=tetha_dot(i,it-1)+dt*tetha_ddot(i,it);
    tetha(i,it+1)=tetha(i,it)+dt*tetha_dot(i,it+1);
    tetha_dot(i,it)=0.5*(tetha_dot(i,it-
1)+tetha_dot(i,it+1));
end

%Record U,V,TETHA
U(i,it+1)=x(i,it+1);
V(i,it+1)=y(i,it+1);
TETHA(i,it+1)=tetha(i,it+1);

%Calculate New Coordinate
X(i,it+1)=X(i,it)+x(i,it+1);
Y(i,it+1)=Y(i,it)+y(i,it+1);

% Constraint Element Supported
X(143,:)=X(143,1);
Y(143,:)=Y(143,1);
Y(177,:)=Y(177,1);

% Calculate Neighborhood Interaction
for j=1:6
    n=T(i,j,it);

```

```

        if n==0
            continue
        end
        r_ij(i,j,it)=(((X(n,it+1)-X(i,it+1))^2 + ((Y(n,it+1)-
Y(i,it+1))^2))^0.5;
        sin(i,j,it)=(Y(n,it+1)-Y(i,it+1))/(r_ij(i,j,it));
        cos(i,j,it)=(X(n,it+1)-X(i,it+1))/(r_ij(i,j,it));

        un(i,j,it)=-((U(i,it)-U(n,it))*cos(i,j,it))-((V(i,it)-
V(n,it))*...
            sin(i,j,it));
        if i==36
            us(i,j,it)=((U(i,it)-U(n,it))*sin(i,j,it))-((V(i,it)-
V(n,it))*...
                cos(i,j,it))+(r1*TETHA(i,it))+(r2*TETHA(n,it));
        elseif i==71
            us(i,j,it)=((U(i,it)-U(n,it))*sin(i,j,it))-((V(i,it)-
V(n,it))*...
                cos(i,j,it))+(r1*TETHA(i,it))+(r2*TETHA(n,it));
        elseif i==107
            us(i,j,it)=((U(i,it)-U(n,it))*sin(i,j,it))-
((V(i,it)-V(n,it))*...
                cos(i,j,it))+(r1*TETHA(i,it))+(r2*TETHA(n,it));
        elseif i==142
            us(i,j,it)=((U(i,it)-U(n,it))*sin(i,j,it))-
((V(i,it)-V(n,it))*...
                cos(i,j,it))+(r1*TETHA(i,it))+(r2*TETHA(n,it));
        else
            us(i,j,it)=((U(i,it)-U(n,it))*sin(i,j,it))-
((V(i,it)-V(n,it))*...
                cos(i,j,it))+(r2*(TETHA(i,it)+TETHA(n,it)));
        end

    if it==1
        if i==36
            fn(i,j,it)=0+kn1*un(i,j,it);
            fs(i,j,it)=0+ks1*us(i,j,it);
        elseif i==71
            fn(i,j,it)=0+kn1*un(i,j,it);
            fs(i,j,it)=0+ks1*us(i,j,it);
        elseif i==107
            fn(i,j,it)=0+kn1*un(i,j,it);
            fs(i,j,it)=0+ks1*us(i,j,it);
        elseif i==142
            fn(i,j,it)=0+kn1*un(i,j,it);
            fs(i,j,it)=0+ks1*us(i,j,it);
        else
            fn(i,j,it)=0+kn2*un(i,j,it);
            fs(i,j,it)=0+ks2*us(i,j,it);
        end
    end

```

```

        end
    else
        if i==36
            fn(i,j,it)=fn(i,j,it-1)+kn1*un(i,j,it);
            fs(i,j,it)=fs(i,j,it-1)+ks1*us(i,j,it);
        elseif i==71
            fn(i,j,it)=fn(i,j,it-1)+kn1*un(i,j,it);
            fs(i,j,it)=fs(i,j,it-1)+ks1*us(i,j,it);
        elseif i==107
            fn(i,j,it)=fn(i,j,it-1)+kn1*un(i,j,it);
            fs(i,j,it)=fs(i,j,it-1)+ks1*us(i,j,it);
        elseif i==142
            fn(i,j,it)=fn(i,j,it-1)+kn1*un(i,j,it);
            fs(i,j,it)=fs(i,j,it-1)+ks1*us(i,j,it);
        else
            fn(i,j,it)=fn(i,j,it-1)+kn2*un(i,j,it);
            fs(i,j,it)=fs(i,j,it-1)+ks2*us(i,j,it);
        end
    end

    FN(i,j,it)=fn(i,j,it);
    FS(i,j,it)=fs(i,j,it);
    fX(i,j,it)=(FN(i,j,it)*cos(i,j,it))-
    (FS(i,j,it)*sin(i,j,it));

    fY(i,j,it)=(FN(i,j,it)*sin(i,j,it))+(FS(i,j,it)*cos(i,j,it));
end

FX(i,it+1)=(sum(fX(i,:,it)));
FY(i,it+1)=(sum(fY(i,:,it)));
if i==36
    M0(i,it+1)=(sum(FS(i,:,it)))*(-r1);
elseif i==71
    M0(i,it+1)=(sum(FS(i,:,it)))*(-r1);
elseif i==107
    M0(i,it+1)=(sum(FS(i,:,it)))*(-r1);
elseif i==142
    M0(i,it+1)=(sum(FS(i,:,it)))*(-r1);
else
    M0(i,it+1)=(sum(FS(i,:,it)))*(-r2);
end

if mod(it,2)==0
    figure(1)
    clf
    hold on
    rectangle('Position',[0 0 1 h])
    axis([r2-2 (2*r2*n_col)+2 -(r2*n_row) (2*b2*n_row+1)])
% axis auto

```

```

        for n=1:177
            if n==36

circles4(X(n,it),Y(n,it),r1,'vertices',6,'color','blue','rot',90)

text(X(n,it),Y(n,it),num2str(n),'FontSize',24*r1,'HorizontalAlignmen
t','center')
                elseif n==71

circles4(X(n,it),Y(n,it),r1,'vertices',6,'color','blue','rot',90)

text(X(n,it),Y(n,it),num2str(n),'FontSize',24*r1,'HorizontalAlignmen
t','center')
                elseif n==107

circles4(X(n,it),Y(n,it),r1,'vertices',6,'color','blue','rot',90)

text(X(n,it),Y(n,it),num2str(n),'FontSize',24*r1,'HorizontalAlignmen
t','center')
                elseif n==142

circles4(X(n,it),Y(n,it),r1,'vertices',6,'color','blue','rot',90)

text(X(n,it),Y(n,it),num2str(n),'FontSize',24*r1,'HorizontalAlignmen
t','center')
                else

circles3(X(n,it),Y(n,it),r2,'vertices',6,'color','blue','rot',90)

text(X(n,it),Y(n,it),num2str(n),'FontSize',26*r1,'HorizontalAlignmen
t','center')
                end
            end

        end

        % clf
        end
        clf
        disp(num2str(it))
    end
end

```

## Circle

```
function [h] = circles4(x,y,r1,varargin)
%% Check inputs:
assert(isnumeric(x),'Input x must be numeric.')
assert(isnumeric(y),'Input y must be numeric.')
assert(isnumeric(r1),'Input r1 must be numeric.')

if ~isscalar(x) && ~isscalar(y)
    assert(numel(x)==numel(y),'if neither x nor y is a scalar, their
dimensions must match.')
end
if ~isscalar(x) && ~isscalar(r1)
    assert(numel(x)==numel(r1),'If neither x nor r1 is a scalar,
their dimension must match.')
end
if ~isscalar(r1) && ~isscalar(y)
    assert(numel(r1)==numel(y),'If neither r1 nor y is a scalar,
their dimension must match.')
end

%% Parse inputs:

% Define number of points per circle:
tmp =
strcmpi(varargin,'points')|strcmpi(varargin,'NOP')|strcmpi(varargin,
'corners')|...
    strcmpi(varargin,'vert',4);
if any(tmp)
    NOP = varargin{find(tmp)+1};
    tmp(find(tmp)+1)=1;
    varargin = varargin(~tmp);
else
    NOP = 1000; % 1000 points on periphery by default
end

% Define rotation
tmp = strcmpi(varargin,'rot',3);
if any(tmp)
    rotation = varargin{find(tmp)+1};
    assert(isnumeric(rotation)==1,'Rotation must be numeric.')
    rotation = rotation*pi/180; % convert to radians
    tmp(find(tmp)+1)= 1;
    varargin = varargin(~tmp);
else
    rotation = 0; % no rotation by default.
end

% Be forgiving if the user enters "color" instead of "facecolor"
tmp = strcmpi(varargin,'color');
```

```

if any(tmp)
    varargin{tmp} = 'facecolor';
end

%% Begin operations:

% Make inputs column vectors:
x = x(:);
y = y(:);
r1 = r1(:);
rotation = rotation(:);

% Determine how many circles to plot:
numcircles = max([length(x) length(y) length(r1) length(rotation)]);

% Create redundant arrays to make the plotting loop easy:
if length(x)<numcircles
    x(1:numcircles) = x;
end

if length(y)<numcircles
    y(1:numcircles) = y;
end

if length(r1)<numcircles
    r1(1:numcircles) = r1;
end

if length(rotation)<numcircles
    rotation(1:numcircles) = rotation;
end

% Define an independent variable for drawing circle(s):
t = 2*pi/NOP*(1:NOP);

% Query original hold state:
holdState = ishold;
hold on;

% Preallocate object handle:
h = NaN(size(x));

% Plot circles singly:
for n = 1:numcircles
    h(n) = fill(x(n)+1.2*r1(n).*cos(t+rotation(n)),
y(n)+1.2*r1(n).*sin(t+rotation(n)), 'r', varargin{:});
end

% Return to original hold state:
if ~holdState

```



```

        hold off
    end

    % Delete object handles if not requested by user:
    if nargin==0
        clear h
    end

end

```

### Element 1\_m

```

function [n_col,n_row] = element_1(l,h,a,b)
%Calculate Possible Configuration & Number of Particles
%   l = length of beam
%   h = height of beam
%   n_col = must be integer
%   n_row = must be integer

n_col=floor(l/a);
n_row=floor(h/b);

end

```

### element 2\_m

```

function [l,h] = element_2(n_col,n_row,a2,b2)
%Calculate Length & Width of Beam

l=a2*n_col;
h=(b2*(n_row))+3*b2;

end

```

### Element\_properties.m

```

function [kn1,ks1,kn2,ks2,m1,m2,I1,I2] =
element_Prop(E,v,del,r1,r2,rho)
%Calculate Stiffness Contact and Mass of Particle

kn1=((2*(3^0.5)*r1)/(3*(r1+r2)))*(E*del)/(1-v);
ks1=(r1/(r1+r2))*((E*del)/(1+v));
kn2=((2*(3^0.5)*r2)/(3*(r2+r2)))*(E*del)/(1-v);
ks2=(r2/(r2+r2))*((E*del)/(1+v));

```

```

m1=(6*(r1^2)*(1/3)*(3^(0.5)))*rho*del;
I1=0.5*m1*(r1^2);
m2=(6*(r2^2)*(1/3)*(3^(0.5)))*rho*del;
I2=0.5*m2*(r2^2);

```

```

end

```

## Initial Position

```

function [X,Y] =
InitPosition01(n_element,h,r1,r2,n_col,n_row,a2,b2,nt)
%Calculate Initial Coordinates of Each Particle

%Create Zero Matrix for X & Y
X=zeros(n_element,nt+6);
Y=zeros(n_element,nt+6);

%Generate X & Y Coordinates for Each Particle

for i=1:n_row
    if i==2
        for j= 1:n_col+3;
            if j==1
                X((i*j)+(i-1)*(n_col-j),:)=r1;
            elseif j==n_col+1
                X((i*j)+(i-1)*(n_col-j),:)=68*r2+3*r1;
            else
                X((i*j)+(i-1)*(n_col-j),:)=(j-1)*a2;
            end
        end
        if i==2
            Y((i*j)+(i-1)*(n_col-j),:)=h-((2*b2)+(0.5*b2));
        else
            Y((i*j)+(i-1)*(n_col-j),:)=h-((b2)+((i-
2)/2)*(4*b2)+(0.5*b));
        end
    end
    elseif mod(i,3)==0
        for j=2:n_col+3
            if j==2
                X((i*j)+(i-1)*(n_col-j),:)=r2;
            else
                X((i*j)+(i-1)*(n_col-j),:)=r2+(j-1)*a2;
            end
            Y((i*j)+(i-1)*(n_col-j),:)=h-(b2+((floor(i/2))*3*b2));
        end
    elseif mod(i,4)==0
        for j=2:n_col+3
            if j==2
                X((i*j)+(i-1)*(n_col-j),:)=r1;

```

```

elseif j==n_col+2
    X((i*j)+(i-1)*(n_col-j),:)=68*r2+3*r1;
else
    X((i*j)+(i-1)*(n_col-j),:)=2*r2+(j-1)*a2;

    end
    if i==4
        Y((i*j)+(i-1)*(n_col-j),:)=h-((0.5*b2)+(5*b2));
    end
end
elseif mod(i,5)==0
for j=3:n_col+3
    if j==2
        X((i*j)+(i-1)*(n_col-j),:)=r2;
    else
        X((i*j)+(i-1)*(n_col-j),:)=3*r2+(j-1)*a2;
    end
    Y((i*j)+(i-1)*(n_col-j),:)=h-(b2+((floor(i/2))*3*b2));
end
else
    for j=1:n_col+3
        if j==1
            X((i*j)+(i-1)*(n_col-j),:)=r2;
        else
            X((i*j)+(i-1)*(n_col-j),:)=r2+(j-1)*a2;
        end
        Y((i*j)+(i-1)*(n_col-j),:)=h-(b2+((floor(i/2))*3*b2));
    end
end
end
end

```

**%Remove Zero Particle**

```

zeroX=X(:,1)==0;
X(zeroX,:)=[];
zeroY=Y(:,1)==0;
Y(zeroY,:)=[];

```

**end**

## Neighborhood

```

function [T] = Neighborhood(n_element,X,Y,r2,nt)
%Calculate Neighborhood Based on Particle Coordinates

%Create Zero Matrix for Neighbor Member
T=zeros(n_element,6,nt+6);

%Find All Possible Member

```

```

for i=1:n_element
    neighbor=(find(X(:,1) >= X(i,1)-2*r2 & X(:,1) <= X(i,1)+2*r2 &
Y(:,1) >= Y(i,1)-2*r2 &...
    Y(:,1) <= Y(i,1)+2*r2));
    s=size(neighbor);
    for j=1:s
        T(i,j,:)=neighbor(j);
    end
end

%Eliminate Main Particle from Member List
for i=1:n_element
    for j=1:6
        if T(i,j,')==i
            T(i,j,:)=0;
        end
    end
end

end

```

*(halaman ini sengaja dikosongkan)*